Classroom Computer News

NOVEMBER-DECEMBER, 1980 VOLUME 1, NO. 2 $2.00

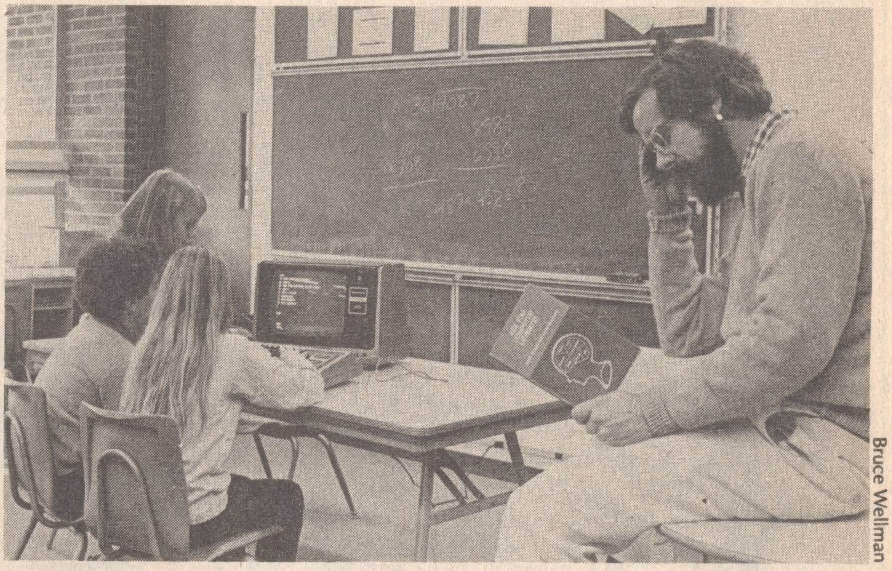
MICRONEWS

• The Japanese are coming . . . with low-cost, high-capacity disk drives that portend impressive storage capacities for future education-oriented computers. The drives, introduced by Fujitsu, a $2.5 billion company, store 84 megabytes and cost $3795. This is 16 times the amount of information in *Webster's Seventh New Collegiate Dictionary.*

• Wang Laboratories, major manufacturer of small computers, has had problems locating programmers so it's started its own graduate school. The Wang Institute of Graduate Studies, Tyngsborough, Mass., opened recently to applicants from across the country. The Institute, says Dean Caroline Wardle, plans "to remain small and select." Curriculum blends academic, management and technical subjects.

• PLATO, well-known educational software package for time-sharing systems, may soon be available for Apple computers. Apple's Glen Polin says several people are now converting PLATO programs. This, he feels, will greatly expand the educational software available for the Apple.

CONTENTS



Bruce Wellman

Computer literacy — for both students and teachers — will be the key to successful educational use of computers during the 1980s.

Computer Literacy: What Should Schools Be Doing About It?

by Daniel H. Watt

The term computer literacy is coming into widespread use without a clear, generally agreed upon definition. I think that the concept remains poorly defined for several reasons. First, the everyday concept of literacy is not clearly defined, having different meanings in different contexts. Second, computer technology is proliferating so rapidly in all phases of our lives that any particular notion of the *specific* knowledge necessary to be computer literate tends to be surpassed as quickly as it is formulated. Third, now that there is widespread agreement that computer literacy is of fundamental national importance, many people with diverse goals are trying to attach that label to their own particular ideas. Yet we need to formulate a concept of computer literacy that we can use to develop the new educational programs we will require during the next few years.

Defining Literacy

My definition of computer literacy is based on an interpretation of the common meaning of literacy. Dictionary definitions of literacy usually include phrases such as "the ability to read and write" and "the state of being well informed, educated." These definitions include no specific standards. The concept of what it means to be literate can vary from culture to culture, and from group to group within a given culture. Literacy has to be viewed as a continuum, from a minimal ability to read news headlines at one extreme to the literacy skills of a professional writer or skilled academic at the

Please turn to page 26.

Classroom Computer News

EDITOR
Lloyd R. Prentice

ASSOCIATE EDITOR
Laurie Beckelman

FEATURES EDITOR
Phyllis Caputo

CONTRIBUTING EDITORS
Jeffrey Katz
Jeffrey Nilson

REGIONAL CONTRIBUTORS
James Dunion (Tennessee);
William D. Hedges (Florida);
Tom Lee (Michigan);
Harry B. Miller (Louisiana);
William J. Wagner (California)

PRODUCTION MANAGER
Lisa Underkoffler

ART DIRECTOR
Mary Rothenbuehler

ASSISTANTS
Andrea Sarrasin
Caron Schwartz

PUBLISHER
Lloyd R. Prentice

DIRECTOR OF
ADVERTISING SALES
Douglas Wales

OFFICE MANAGER
Laura Koller

Classroom Computer News is published by:
Intentional Educations
80 Brighton Avenue
Allston, Massachusetts 02134

Subscriptions: 1 year $9.00.
Please send all subscription correspondence to:
Classroom Computer News,
Box 266, Cambridge,
Massachusetts 02138

Advertising: Please send all advertising inquiries to
Douglas Wales, *Classroom Computer News*, Box 266,
Cambridge, Massachusetts 02138;
617/787-0240.

A Machine for the Whole Human Being

by Lloyd R. Prentice

Recently, a publisher recounted some prejudices that he has encountered among buyers of educational software.

"... many teachers dislike programs that use the game paddles — too much like home video games, I guess. They seem to regard programs that communicate through the keyboard as more serious of purpose.

"Also, they dislike sound. They say it disrupts the class.

"Color is okay, as long as it doesn't detract from the message ..."

Communicating with Computers

The issue here is how people should talk to computers and vice versa. At stake is the ultimate success of the computer as an educational tool. The question is whether the prejudices confronted by the publisher are rooted in sound pedagogy or whether they arise out of an attempt to make a unique, new medium resemble the old, comfortable and familiar.

One can argue that the computer is one of the most versatile communication tools ever invented. It combines in one medium some of the best features of the blackboard, the textbook, the teaching machine, the animated film, the library, the film strip, the phonograph, the video system, the telephone, the telegraph, the typewriter, the sketch pad and the music synthesizer. It can enhance human communication at many levels — as a medium for self-expression and one-to-one interaction, as a resource for group problem solving or as a network controller for linking up millions of people across vast distances.

Indeed, the computer can receive, store, process, generate or send text, pictures, voice, music, mechanical motion — any symbol or signal that can be translated into electrical form and back again. This means that the computer can facilitate human communication through visual, aural and tactile channels. It wouldn't even be farfetched to hear that some creative engineer has employed the computer to facilitate communication through the gustatory or olfactory modes. In fact, Stanford Research Institute has already used the computer in experiments to bypass the senses altogether in an effort to communicate directly with the human brain.

Decisions and Dilemmas

The very versatility of the computer as a communication device imposes upon the designer a bewildering array of choices. Market economics compels the designer of a particular machine to implement but a few of the many possible ways of passing information between user and computer. Some of the methods that have been chosen in the past include switchboard-like patch panels, toggle switches, punched cards, punched tape, touch-sensitive screens, direct voice entry, blinking lights, cathode ray tubes, various forms of printers and plotters, and synthesized human speech.

The designers of recent-vintage small computers marketed for home and school use were under enormous pressure to minimize costs and maximize ease of communication. They chose the standard typewriter keyboard and the game paddle as basic input devices and the video screen and the inexpensive dot-matrix printer for output. As added features, several manufacturers implemented color video and sound capabilities. The suitability of the computer for game playing was a strong criterion in the design of several of these machines — particularly the systems brought out by Atari and Mattel.

The queston is whether or not the current generation of small computers provides communication resources that are optimal for education. The answer is probably not.

The QWERTY keyboard may well be an obstacle for young children. The video display is certainly not the answer for youngsters with visual disabilities. Sound may be useful, but perhaps should be conveyed through earphones rather than broadcast through speakers. Perhaps the money spent on color display

Please turn to page 21

Radio Shack Shows Wares

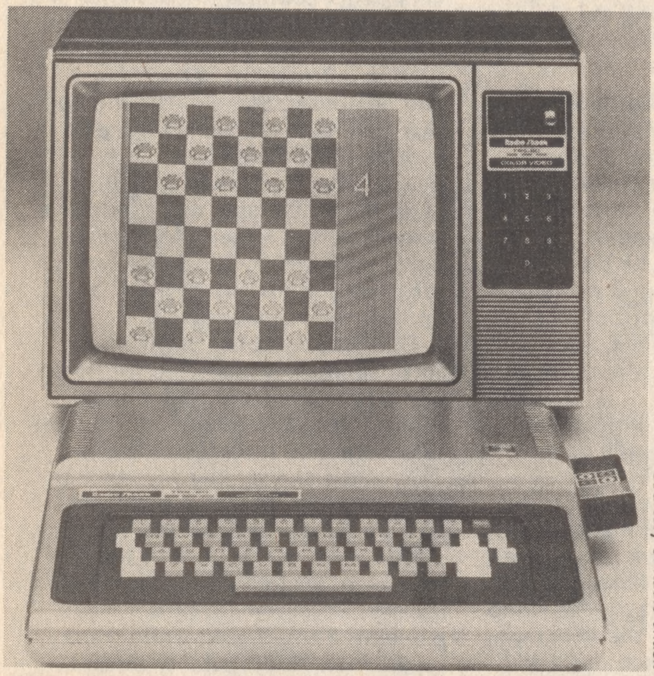
During the last week in September, representatives from Radio Shack, some of whom referred to themselves with a smile as "country boys from Texas," came to the St. Regis in New York to talk about their company's new products.

Steve Leinenger, who had helped create the original TRS-80, filled *Classroom Computer News* in on two of Radio Shack's newest offerings: a hand-held computer and the color TRS-80.

Steve had been in Japan working on the hand-held computer, which a Japanese company will manufacture for Radio Shack. The pocket TRS-80 is faster and easier to use than a slide rule calculator and helps the user be more accurate, Steve told us. Computer users do not have to remember complicated formulas or the order in which to enter numbers as they would with a calculator. Correcting mistakes is also easier. People can look at each number they have entered, find the wrong number and correct it. With a slide rule calculator, they often have to start again.

The BASIC in the pocket computer will do almost anything the table model TRS-80 will do to ten decimal places, except the graphics. Users can define their own math functions or use the built-in functions, which include trig, inverse trig, and log functions as well as commands for finding square roots, angular conversions, integers and absolute values.

As for using the pocket computer in schools, however, one Radio Shack representative, who had taught high school physics, told us, "They're too small and too easy to rip off; but they'd be really great to help teach physics, chemistry, and math if you could solve the security problem."



Courtesy of Radio Shack

Radio Shack's color TRS-80 was one of the new products displayed at a recent meeting in New York.

Radio Shack's color TRS-80 is another matter. The company plans to sell it to schools, now the second largest market for Radio Shack's small computers. Steve Leinenger saw the color TRS-80 used primarily for teaching children programming. The color and sound capabilities and graphics language are ideally suited for that, he said.

The most powerful color TRS-80, which has 16K of RAM and the extended color computer BASIC ROM is $600 without the monitor or cassette and will be available in December. It provides a screen resolution of 256 × 192 with eight colors. Users can define, enlarge, reduce and rotate images; they can move images from one part of the screen to another and draw lines around them. Type BOX and a line becomes a box. SOUND adds tones to programs. Editing, specific error messages and user definable keys are other features.

At present, Radio Shack has not developed a disk system for the color TRS-80, but it is working on one. Asked whether not a disk system might limit the ways teachers can use the color computer for language arts, reading and social studies, one Radio Shack official replied, "You never can tell what we might put in those ROM packs you plug in on the side."

Reader's Digest Acquires *Source*

The Reader's Digest Association, Inc. has acquired a majority interest in Source Telecomputing Corporation, America's pioneering electronic information service.

The Source, which presently has nearly 7,000 subscribers, is generally regarded as the largest interactive computer information service in existence. Using an assigned account number and password, subscribers can connect their personal computers or computer terminals, by telephone, to the mainframe *Source* computers, which hold nearly 2,000 programs and databases. Among other services, *The Source* offers point-to-point electronic mail.

The Reader's Digest acquisition assures expansion of the network. For subscribing educators, networks like *The Source* can provide fast and efficient means of sharing programs and computer application ideas with colleagues anywhere in the country.

FCC Changes Help Rural Schools

A fundamental change of the FCC rules governing WATS and other long-distance telephone service may make it possible for schools located in rural areas to take advantage of on-line "information utilities."

In most major cities in the United States, a local telephone call will connect a microcomputer and modem to one of the major data networks such as Telenet or Tymnet. In rural areas, the added long-distance toll charges needed to call the nearest network node usually make on-line use costs prohibitive.

The changes in the rules, which will take place in March, would enable a number of small users (school districts, small companies, businesses) in the same town to share a WATS line.

Unfortunately, the American Telephone & Telegraph Company, which provides WATS service, apparently anticipated the commission's ruling. In September they proposed a new WATS rate schedule that sharply curtails the high-volume discount offered in the past.

Nevertheless, it is still believed that the rule changes will produce a mechanism to keep rates in line with costs and lower charges to smaller users.

New Course for Jersey Teachers

Elementary and high school students all over Bergen County, New Jersey, will soon be using microcomputers to solve mathematical equations, to compose music and to find books in the school library, thanks to

a new course their teachers are taking.

Fairleigh Dickinson University now offers a series of accelerated workshop sessions designed to develop the teacher's competence in creating individualized instruction programs using the microcomputer. Teachers need no previous computer experience.

The course consists of four full-day sessions once a week, plus a minimum of 15 hours in the computer laboratory for each student. Both elementary and secondary school teachers attended this summer's sessions. By the end of the course they were creating computer programs in mathematics, music, literature, library science, history and social sciences.

One librarian, for example, developed a program through which high school pupils could find all the available school library books on American presidents. Subjects appear on the computer's television screen, enabling the pupil to find biographies, political histories or accounts of some specialized facet of a president's career in just a few minutes.

Dr. Stephen M. Gittleson designed and teaches the course; Silent Partner, a Fort Lee, New Jersey, computer dealer, provides equipment for the computer lab (the microcomputers are Apples).

For further information, please contact: Dr. Stephen M. Gittleson, Fairleigh Dickinson University, 1000 River Road, Teaneck, NJ 07666.

A Fair Break for Youth

A National Fair Break program that will use computer-based educational and vocational training programs to address the problem of unemployment among disadvantaged youths has been announced by Control Data Corporation (CDC). CDC proposes to use its High School Skills curriculum, Basic Skills curriculum and expertise in computer-oriented career counseling to work with community agencies and schools to train one million 16-21 year olds over the next three years. The long-term goal is to place these youths in jobs.


Courtesy of Control Data Corporation

Using its High School Skills curricula, Control Data Corporation plans to train one million underemployed and unemployed youths during the 1980s.

The National Fair Break program expands the concept underlying the Fair Break Centers, first established by CDC in 1978. These centers, which now operate in 40 communities nationwide, use Department of Labor funds to teach basic skills, high school skills and vocational skills to underemployed and unemployed individuals. Many participants are high school dropouts. The centers use CDC educational programs to help these participants earn their high school equivalency degrees, a first step toward helping them find good jobs.

The National Fair Break program will target specific geographic areas for more extensive services. It will work with 16-21 year olds, both in and out of school, and it proposes to provide jobs as well as skills training. A CDC spokesperson says that the program, like the Fair Break Centers, is based on the belief that, "government can't solve the problem of unemployment alone." CDC will take a leadership role in organizing networks of community organizations — social service agencies, schools and businesses — to provide training, counseling and jobs for the disadvantaged youths the program reaches.

Reports Available

- A September 1980 conference sponsored by Computer-Using Educators (CUE), Santa Clara Valley Mathematics Association and California Science Teachers Association featured 40 speakers and workshops addressing the theme, "Classroom Applications of Computer, K-12." CUE has compiled contributions and handouts from almost all the sessions and is offering them for $10 postpaid (outside U. S. and Canada add $5). Send checks payable to Computer-Using Educators to: CUE Fall Proceedings, PO Box 846, Mountain View, CA 94042.
- In May, 1980, Technical Education Research Centers (TERC) completed a study for the Northeast Regional Education Planning Project surveying the use of microcomputers in instruction, K-12. The report of that study includes information about ways microcomputers are being integrated into the classroom in New England and New York, descriptions of programs and information about suppliers, resources, publications, program listings and acquisition policies. Copies are available for $6.00 to cover the cost of duplicating plus postage and handling. Prepayment is required for reports shipped. Send checks payable to TERC to: TERC, 44 Brattle St., Cambridge, MA 02138. Please note #168 on your request and check.
- During the school year 1977/78, four computers equipped with LOGO and Turtle Graphics were installed in an elementary school in Brookline, Mass. All sixth grade students in the school had between 20 and 40 hours of hands-on experience with the computers. The National Science Foundation RISE program funded a very detailed documentation of the learning experiences of a subset (16) of the students chosen to represent the entire span of scholastic abilities — from the 3rd through the 99th percentile on achievement test scores. Two out of a projected set of three volumes reporting this project are now available for $3.50 each: *PART II: Project Summary and Data Analysis* and *PART III: Student Profiles*. Send checks payable to the MIT Artificial Intelligence Laboratory to: LOGO Group, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 545 Technology Square, Cambridge, MA 02139.

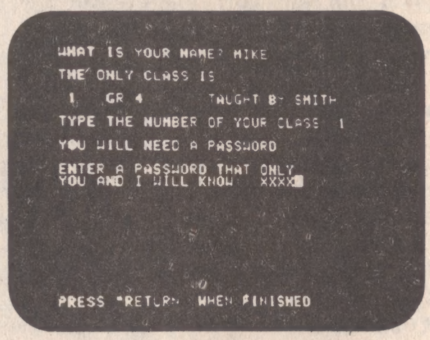
From Hardbacks to Software

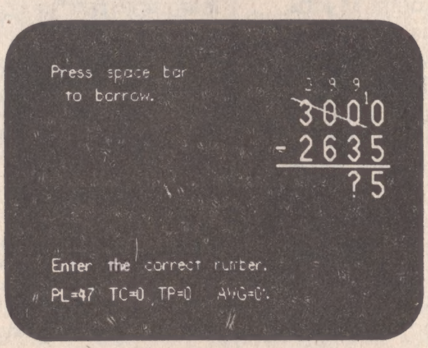
Despite attitudinal and marketing obstacles, many textbook publishers are shelving paper in favor of disks.

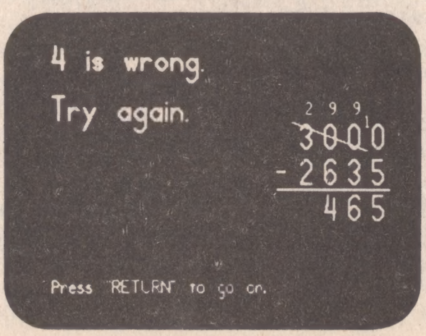
by Phyllis Caputo

Should we or shouldn't we is a question more and more textbook publishers are asking about entering the educational software market. While many still feel that negative teacher attitudes toward computers, technological difficulties and dwindling school budgets make the market too risky, they recognize that educational computing is a force they can no longer ignore. Indeed, of 14 major textbook publishers *Classroom Computer News* recently surveyed, only three reported no active interest in educational software development, while seven said they have made major time and money commitments to the field. (The seven are: Science Research Associates; Addison-Wesley Publishing Company; Houghton Mifflin Company; Bell & Howell; Scott, Foresman and Company; Milliken Publishing Company and Hayden Book Company.)

As for the rest, a Holt, Rinehart, and Winston spokesperson summed up the general sentiment: "It's too soon to make a commitment, yet too late to ignore the whole field." Holt, Rinehart; McGraw-Hill Book Company; Ginn and Company; and Charles E. Merrill Publishing Company are the fence-sitters, all investigating the field without committing major resources. McGraw-Hill, for example, has spent the past 18 months in market research. They said that they're now considering taking a first step by marketing some acquired software (software developed by authors who are not in-house) later this school year.







Caution Explained

These wait-and-see companies cite a number of reasons for their caution. One of the first and foremost is teacher attitude. While some publishers feel that negative teacher attitude toward computers is rapidly being eroded, others feel it is still an obstacle. One technical consultant at Ginn points out that "teachers still feel that computers threaten their jobs. They don't see computers as a tool that can enhance their work by allowing more time for one-to-one interaction."

Once past this psychological barrier, other, more technical, problems arise. One is the question of software compatibility with the various microcomputers on the market. Should the publishing company tie itself to one of the microcomputers on the market? Or, should it produce several versions of its software for the different microcomputers? And what of acquired software classroom teachers produce? Can the publisher be sure that the software will transfer out of the microcosm of that particular teacher's classroom?

Cost is a third factor behind the caution. Bob Bowen, vice president of the school division of McGraw-Hill Book Company, predicts that "the cost of *good* software is going to be very high." Frederick Michael, microcomputer marketing manager for education at Bell & Howell, adds that "the fact that there just aren't enough programmers" will further compound the problem. This situation is endemic to all companies using computers. One final factor in the high cost of producing educational software is royalties. Sometimes both authors and programmers contribute to a software product and equal royalties need to be paid. Publishers are all too familiar with copyright problems that can emerge when more than one person works on a product.

And finally, even while looking at excellent projections for the microcomputer market, publishers face the reality of budget cutbacks in school systems throughout the country. How can they justify an untried new venture to stockholders when established markets are not exactly booming? And, always in the back of everyone's mind is the experience with computers in the schools in the sixties. Are computers really here to stay in the classroom?

A Range of Solutions

Publishers who have made the

commitment to educational computing are surmounting these problems in various ways. Take the problem of system compatibility — some publishers are producing software exclusively for one of the microcomputers on the market. Scott, Foresman software runs on the Texas Instruments 99 microcomputer. In fact, computer and software can be bought as a package. Bell & Howell has adopted a similar strategy. Its Genis software runs on an Apple microcomputer specially modified at Bell & Howell.

Other companies choose to produce several different versions of the same program. Milliken's *Math Sequences* software program, for example, is produced in Apple II Plus, TRS-80 and Commodore PET versions. Some SRA software is available in both Atari and Apple versions.

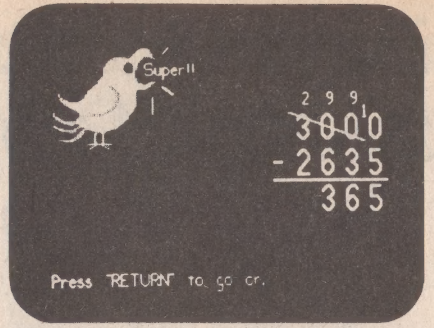
Houghton Mifflin has chosen yet another solution to the problem of software compatibility. Both its *Stride* and *The Answer* programs make use of a dedicated computer, one expressly devoted to running that software.

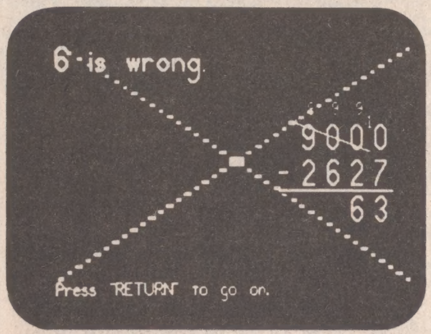
Some publishers have been able to keep down the costs of producing software by working within the traditional model for developing text materials. While the actual process may differ to some degree from publisher to publisher, the essential elements are the same.

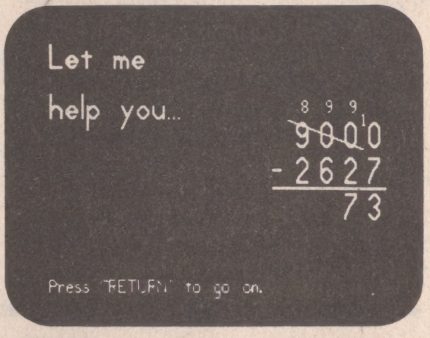
Scott, Foresman's software development process serves as a good example. The process begins when an author proposes an idea. The author may be an expert in his curriculum field and/or someone experienced with computers. In the next step, *courseware development specialists* (usually subject matter generalists Scott, Foresman hires) work in conjunction with artists and programmers to develop the final product. Once all the revision work is finished, work on the prototype begins. The prototype is then sent to test sites. Teachers and administrators at the sites are interviewed for reactions and suggestions. The product is modified as needed and is then ready for marketing.

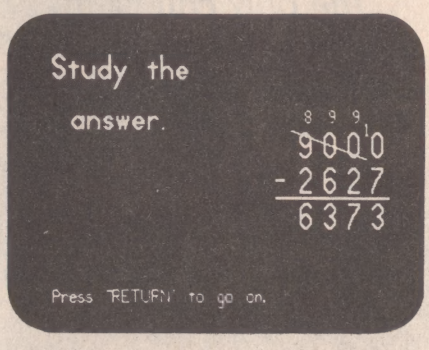
The new element in the traditional process of curriculum development is the addition of technical experts. In some cases, professional programmers, and in others, teachers with strong technical backgrounds and/or programming skills, are brought in.

A few companies began their software ventures by converting existing text or time-share materials into microcomputer versions. In this way, they were able to market an already proven product in a new medium.









Courtesy of Milliken Publishing Company

Milliken's original *Math Sequences* package was based on Computer Curriculum Corporation's *Math Strands Program*. The new Milliken *Math Sequences* package, however, was primarily produced by an in-house contingent of curriculum specialists, editors and programmers, as will be forthcoming programs.

Quite a few companies have tied their software to existing print products. Addison-Wesley has developed a microcomputer math series which correlates with its standard math sequences. Houghton Mifflin's *The Answer* is a classroom management system for its successful kindergarten through eighth-grade series.

Teacher Attitudes

The issue of negative teacher attitude toward computers is one which publishers have just begun to face. Peter Cuseck of Ginn's Strategic Planning Division feels that "more teachers taking computer literacy courses will help the problem to some extent." Richard Monnard, vice president and editorial director at Addison-Wesley, believes that "the computer can be presented to the teacher as a tool to better manage bigger classes." Other publishers, such as Bell & Howell, provide some in-service training with their new software and/or systems, and many others believe that teacher attitudes toward computers will change as the society becomes more computer literate.

Educational publishers have committed themselves to educational computing and have found creative solutions to the problems that confront *electronic publishing* (producing products and/or programs for computer use). But the situation is such a fluid one that next year's picture will undoubtedly be different from today's. An exciting area is opening up. The number of companies involved in electronic publishing will certainly increase.

What the Publishers Are Doing

Addison-Wesley Publishing Company

Addison-Wesley has been working on various math packages which they will introduce to the National Council of Teachers of Mathematics in April 1981. The programs should be available by next fall.

One package is designed for teaching high school students algebra and analysis; the other is a junior high school and upper elementary package of games and activities that promote arithmetic skills. Both series

Please turn to page 21

Buying Big: Purchasing Prowess Texas Style

Texas's Region IV Education Service Center recently purchased 600 microcomputers, and the way they did it may well provide a model for others considering buying big.

by Phyllis Caputo

The Region IV Education Service Center in Texas has recently awarded Bell & Howell a contract to supply microcomputers to the 101 school districts the center services. The bidding and evaluation process preceding this award may very well revolutionize the process by which schools buy microcomputers in the future.

"The Region IV people really challenged the hardware providers. They took some people to task," says Frederick Michael, Bell & Howell microcomputer marketing manager for education. Bidders were asked to extensively document their hardware, operational software, educational programs and in-service training for teachers in response to 50 pages of specifications.

The Texas State Department of Education divides the state into 20 geographic regions, each of which is served by an education service center. These centers were originally set up under Title III to help keep schools abreast of educational change. They now provide a wide array of services, from administrative to classroom support. For the Region IV Education Service Center, this includes assisting its school districts with educational computing. Region IV, one of the smaller regions geographically, has a population of 800,000 students, roughly one-third of Texas's total student population. Region IV serves 101 school systems, one of which is the Houston Independent School District.

The Background

A time-share system consisting of twin Control Data Corporation CYBER 72 minicomputers links Region IV's districts. Seven hundred and twenty time-share terminals hook into the central computers.

According to Pat Sturdevant, coordinator of computer-based instruction at the Region IV center, "the number of terminals was stabilizing in the year prior to the microcomputer bid due to high cost. Microcomputers, on the other hand, were cost effective and could be worked into the existing distributor network."

Two other considerations made the Region IV center take a closer look at buying microcomputers. First, individual schools were already buying them on their own. The schools then expected the center to support the equipment. This lead to what Pat Sturdevant calls "an unbelievable dilution of resources." The center saw the need to standardize equipment so that it could coordinate resources in offering assistance. Second, the center was already assisting the schools in hardware procurement. "There was no point in school districts doing it on their own since the center could negotiate better costs, insure in-service training support, get better maintenance terms and get more software coordination for the region," said Sturdevant.

Cost was not the only criteria in awarding the contract — software, maintenance and in-service training were all important considerations.

As a result, the center appointed a committee in December 1978 to study the state of the art in the microcomputer field and to make recommendations. The committee, which included center staff, data processing specialists, superintendents and representatives from the area school districts, met for four months during which its members discussed issues including standardizing and maintaining equipment, coordinating software and high volume purchasing. The committee also considered instructional and management applications for the microcomputers and the need for computer literacy programs for teachers in the region. The committee concluded that purchasing micros would be premature because of the lack of educationally sound software.

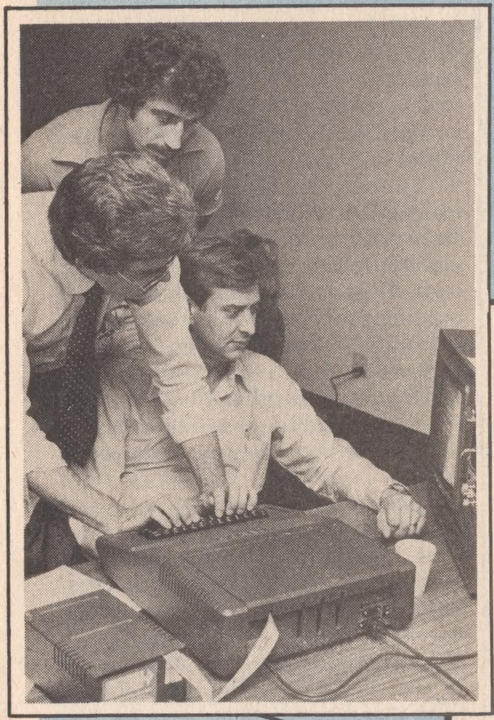
In January 1980, the committee was reactivated. At this time, its members decided to buy in high volume and to aim at standardizing equipment. Computer companies were asked to submit proposals in response to a 50-page list of specifications.

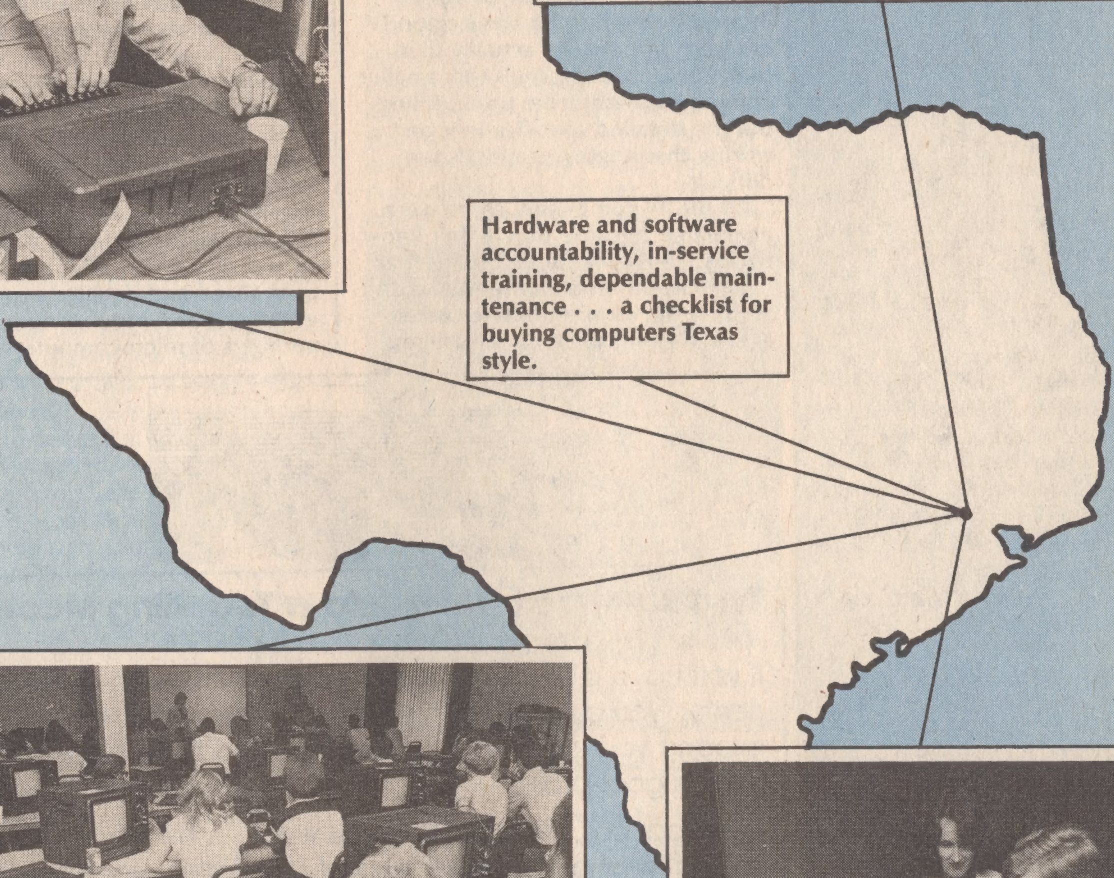
The Bidding Process

These remarkably detailed specifications are what make the Region IV story so significant. Cost was not the only criteria in awarding the contract. "When you compare the microcomputers for price," says Pat Sturdevant, "there is not much difference." In addition, Region IV emphasized the availability and quality of educational software. In the weighting system established by the committee, cost and software were given equal weight. The weighting factors broke down as follows:

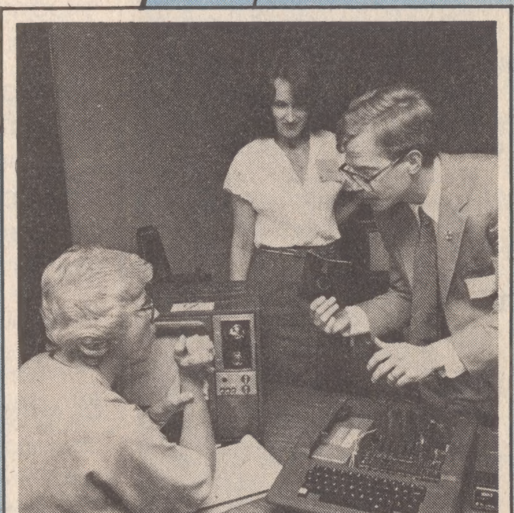
educational software	30%
cost	30%
hardware and operational software	25%
service and support	15%

Another innovation was the





Hardware and software accountability, in-service training, dependable maintenance . . . a checklist for buying computers Texas style.



Photos courtesy of the Region IV Education Service Center

detailed manner in which hardware providers had to account for these areas. For example, they had to key their equipment specifications to 150 separate items in the original bid and had to provide supporting documentation.

The same exactness of detail applied to software. Each provider had to prepare a complete list of educational software that could run on its microcomputer, including program name, description, price, and the name and phone number of a contact person.

Maintenance was given less weight but was also an important consideration. Pat Sturdevant said that Region IV wanted to eliminate the situation in which the hardware manufacturer delivers a microcomputer and then steps out of the picture. Region IV wanted more service. It gave potential providers a map showing the areas in Region IV and their diverse needs and asked them to propose a range of service options including on-site and walk-in service.

Training was handled similarly. Manufacturers provided specific lists of materials available for teacher training. These included both materials developed in-house and others that the manufacturer knew could be used for training.

The Response

Forty-four companies expressed interest in bidding for the Region IV contract, but only 14 actually filed bids. Pat Sturdevant feels that smaller companies might have found filling out the detailed specifications and writing the lengthy proposals too difficult.

Of the 14 companies, some were eliminated because they didn't know what software was available for their machines, others because they didn't provide the required service coverage. Provisions for teacher training were another stumbling block — some training programs were primarily aimed at the industrial market and were therefore inappropriate for use in public schools.

Bell & Howell's bid "was not the lowest, but was the best," says Sturdevant. Bell & Howell, which manufactures a version of the Apple microcomputer, offered the most comprehensive selection of educational software and was able to provide in-service training and easy maintenance through a pre-existing service network. Region IV schools will house 600 Bell & Howell microcomputers as a result of this long bidding process. By mid-summer, 226 units had already been installed and connected to the existing timeshare system; another 20 or 30 units are now in. Bell & Howell dealers will service the machines if the need arises.

Since June, 50 organizations, including the Department of Defense and several state departments of education, have contacted Region IV for copies of the contract specifications. As more schools acquire microcomputers, Region IV's approach may provide a model for a centralized and detailed process for evaluating and purchasing large numbers of microcomputers.

We Want Your Words

Are you an educator with a nose for computer news?

If so, *Classroom Computer News* would like to publish your words.

In terms of content, we are interested in stories that help teachers and educators make better use of computers in the schools.

In terms of style, we are interested in news and feature stories that are fact filled, well-written and expressly directed toward our bright and critical readers. Stories should be lively, concrete and written for intelligent humans, with minimal use of computer or educational jargon.

News stories, as presented in the On Line section, run 300 to 400 words. Feature stories run from 800 to 2500 words. Clear, high-contrast glossy 8 x 10 photos are welcomed. Computer programs must be well-documented, thoroughly tested and machine printed. Use a new ribbon and clean white paper for program listings. Manuscripts should be typed and triple-spaced.

Classroom Computer News will pay $20 to $50 a printed page upon publication for reader-contributed stories, depending upon the quality of the manuscript and the amount of editing/rewriting needed. Contributors will also be given five copies of the issue containing their submission.

Send queries or manuscripts to Lloyd R. Prentice, Editor, *Classroom Computer News*, Box 266, Cambridge, MA 02138. Allow six weeks for acknowledgement. All submissions will be acknowledged, but only manuscripts accompanied by a self-addressed stamped envelope will be returned if not purchased.

A School Administrator Looks at VisiCalc (and Likes What She Sees)

by Amy Jane D. Winchell

School administrators who formulate budgets and manipulate other kinds of data will find an extraordinarily useful tool in Personal Software's *VisiCalc* program.

Imagine a large sheet of paper divided into 63 columns across the top and 254 rows down the side. At the top of as many columns as you need, write headings (category, fiscal year and so forth). Down the side, identify as many rows as you need — account number, item, sub-totals, etc. But instead of a paper worksheet, this layout is the *VisiCalc* program for an Apple microcomputer. Rather than write in the headings and label the rows on a paper worksheet, you type the information on the microcomputer keyboard to set up an "electronic worksheet."

Now enter the data you are using *and the calculations you want performed with that data.* At every entry point, you can type in the formula to be used to derive that number: the ratio to be applied, the addition to determine a total, the percentage to be calculated — whatever you would be doing if you were using your calculator to compute the figure. If, for example, you would be calculating the number in column G, row 18, by adding the numbers in column G, rows 16 and 17, you just type in the value (V) G16+G17 in the G18 space. Describing this process is much more cumbersome than doing it. The procedure is as straightforward as hand calculating, and can become quite rapid with practice.

Easy Changes

When the format for your project is completed, you're all set. From now on, the microcomputer will do the work, with dazzling speed and accuracy. It will immediately display the results of its calculations, of course, including that all-important "bottom line." But now comes the good part: you can change any of the numbers in your entry data and all of the related figures are automatically recalculated. Move the cursor indicator to the data you want to change, enter the new data and presto! With what seems like the speed of light, the computer revises the calculations and displays the new results for your contemplation. Bottom line too high? Increase a percentage cut. Bond amount increased? Back to the top line for a revised entry. You can try all kinds of data manipulations to test the outcomes of possible changes or adjustments for different items, to assess the budgetary effects of administrative options or to produce whatever results you want to see.

If you are working with a lot of columns and rows, the video monitor of course cannot display everything at once. You have to select the sections you want to see at one time. But you can split the screen vertically to put non-adjacent sections next to each other, or split it horizontally to observe the effects of top-line data changes on the bottom line.

Suppose you discover that you omitted a row. Or you have to enter a new line item that has been introduced since you laid out the program — a fairly common occurrence. No problem. Move the cursor to where you want to add the line, enter the new data and ZIP! All the subsequent rows are renumbered (and recalculated) for you.

To get a "hard copy" of your data, converting back from the electronic worksheet to a paper worksheet, you need a computer printer attached to the microcomputer. The bigger and faster the printer, naturally, the more of the program you can print out at once. But you can operate effectively even with a small thermal printer. Just run a couple of columns at a time on the printer and paste up the complete program. It still beats having to type reams of hand-written data, which is not only time consuming, but also vulnerable to human error.

A Vote of Confidence

As a school administrator who has done some of these functions the hard way, I am struck by what a useful tool *VisiCalc* can be. I can accomplish calculations that have taken hours in minutes; information that came back in days can be available immediately.

Think of having the program at your elbow through the budgeting process. Line item changes, additions and deletions, percentage reductions — all the revisions that take place from initial formulation through final adoption (and subsequent modifications, budgeting being a year-round process) could be entered on the spot. You'd see their immediate effects on the overall budget and the new bottom line.

Consider having *VisiCalc* available for collective bargaining sessions: you could enter percentage changes in rates of increase, changes in ratios, effects of changing the public share of insurance costs and so forth, enabling you to immediately evaluate the impact of such changes on a total package. Or imagine using *VisiCalc* to calculate the effect of a school building bond on a town tax rate. Once you had formatted the program in *VisiCalc*, you could change the amount of the bond, or the tax base or the number of years, and the new data would be calculated and displayed instantly, ready for printing.

I am also impressed that a tool with such potential for increasing administrative efficiency costs little more than a couple of electric typewriters. An Apple II or Apple II Plus microcomputer with 32K memory, a disk drive, and a video monitor (or RF converter to connect to a standard video receiver) will do the job. The *VisiCalc* program itself is about $150. It comes with a detailed instruction booklet. By the time you are through Chapter 2 you will be off and running with your own ideas.

Amy Jane D. Winchell is director of instructional services for the Walpole Public Schools, Walpole, Mass.

Special Tools for Special Needs

Computer-assisted language instruction for deaf students.

by Jeffrey Katz

Picture yourself swimming the English Channel. Now, imagine accomplishing this feat after only a few words of instruction and a stiff push. A difficult task becomes an impossible one. This is precisely the situation in which most deaf students find themselves — swimming through the English language without proper preparation.

Deaf children are often poorly prepared because of the difficult way they must acquire language. "The important difference between hearing and deaf children," suggests Dr. Kirk Wilson of Boston University, "is not necessarily the mode by which they communicate but the drastic difference in total language input." The hearing child receives this input almost effortlessly. But the deaf child experiences only the language communicated directly to him. "Not only is the number of sentences reduced for the deaf child," continues Dr. Wilson, "but his perceptions may be affected by the simplified or distorted manner in which language is often presented to him."

Typically, such constricted language development leaves a seventh- or eighth-grade deaf student struggling at a second-grade reading level. Despite intensive classroom drill, reading and writing are only two of the many courses students must take, so strong English communication skills are too often only partially developed.

Computer-assisted instruction (CAI) promises to alleviate the proverbial problems of teaching grammar and reading skills to deaf children. Researchers are designing programs which provide stimulating learning activities at appropriate reading levels. Two Massachusetts schools, The Learning Center for Deaf Children in Framingham and The Boston School for the Deaf in Randolph, are currently using such programs to meet the unique language needs of their students.

The Learning Center for Deaf Children

There is a waiting line at The Learning Center, and it's not for lunch or recess. It's for a chance to sit at one of eight classroom computer terminals hooked up to a Digital Equipment Corporation PDP-11/34 minicomputer. The school's software, developed by Bolt, Beranek and Newman, Inc. of Cambridge, Mass., in 1977, integrates math, language and reading programs. A daily "menu" offers games and programs ranging from *Spelling Bee* to *Calculator*, all of which adapt to the performance level of each

Jeffrey Katz

A young girl at the Learning Center for Deaf Children uses a touch-sensitive screen.

child.

All software uses a language suited to the students' low English skills. Children initially communicate with the computer through pictures and animation. They can identify objects and create more complex images by merely touching their fingers to the screen. This makes learning a complicated computer language unnecessary. As individual students progress, they carry out more and more communication through language-centered commands. Most of the programs still use animation because of its motivational value, but gradually the children become more proficient at completing interactions without relying on it.

For one of the school's most popular activities, the students are on their own:

TO: Girl
FROM: Boy

Dear ________
You are beautiful!!!!!

Why cute me? Why crazy me? Bowling to come yes or no? Do you want to go and get ice cream Yes or no? I want to go to your house.

Love ________

TO: Boy
FROM: Girl

HI ________
YOU ARE VERY CUTE.
YOU ARE VERY CRAZY.
LOVE ________

Love letters intercepted by a quick-witted teacher? Playground post office games? Actually, these letters are products of the electronic message component of the school's learning program. Students work completely independently at *Mail*, as it's known in the classroom. Each receives his own computer "mailbox" and may spend any free time at the terminals sending letters or checking daily mail. Student autonomy, teachers point out, is one of the features that makes *Mail* so appealing. Another, of course, is the excitement of communication, of writing messages and receiving replies from classmates and teachers, of having a mailbox of one's own.

The results of this extracurricular experiment (*Mail* is not used for regular classroom assignments) have been dramatic. Students who previously had spent little time writing in class and none outside of class, are now writing regularly. In addition, notes teacher Jodi Glicklich, they are composing longer messages and showing more attention to grammar and word order.

What is the future of computer-assisted instruction at The Learning Center? According to Christopher Huggins, coordinator of resources and computer services, the system is experiencing growing pains. The school is seeking financial support with which to develop new instructional materials, to increase disk storage and to program for improved performance. Meanwhile, mailboxes bulge and students line up at the terminals to work on what we know as grammar and reading, and what they know as *Jotto*, *Animal* and *Scramble Word*.

The Boston School for the Deaf.
In Randolph, Mass., at The Boston School for the Deaf, students do battle with the English language side by side a formidable ally called ILIAD (an acronym for the mouth-filling Interactive Language Instruction Assistance for the Deaf.

ILIAD is the prototype of a system that Dr. Kirk Wilson of Boston University's School of Education and Dr. Madeleine Bates of Bolt, Beranek and Newman, Inc. are developing. The minicomputer-based system, supported since 1978 by a grant from the Bureau of Education for the Handicapped, creates a "linguistic playground" that helps students both produce and understand written English.

Deaf students who previously spent little time writing are waiting in line to leave messages on their school's computer.

Student control is central to the ILIAD project's philosophy. "Students work independently," observes Chris Rowe of the Boston School, "and each is responsible for choosing his level of competence. This challenges the students to experiment with new vocabulary and sentence patterns."

Each student makes choices at various stages of the tutorials to determine basic sentence type, vocabulary level and specific task. For example, a student may want to work on changing declarative sentences into questions. He will request complex or simple sentences at a particular vocabulary level, and ILIAD will generate an almost infinite number of different examples.

Jeffrey Katz

A Learning Center for Deaf Children student took time out from his computer work to show the sign for "computer."

What makes ILIAD different from traditional CAI systems for language instruction is this capacity for generating virtually limitless example sentences. While using traditional systems, a student may exhaust the prestored examples before he has completely understood the material. With ILIAD, he can drill until he is confident enough to continue.

Although it's a little too early to judge student performance (ILIAD has been in the classroom less than a year), Ms. Rowe has already observed a much-increased vocabulary in some students. She is enthusiastic about the future of computer-assisted instruction and cites the importance of cooperation between teachers, students and ILIAD project staff in developing a successful program.

In the past, the costs of acquiring and maintaining large systems have frustrated computer-assisted language projects. But according to Dr. Wilson, specific plans have been made to condense the large ILIAD system into a more manageable microcomputer-based system during the next two or three years.

The computer doesn't offer deaf students an effortless sail through the turbulent English language. But it is helping to bring them within view of stronger English communication skills and the experience of total communication.

Jeffrey Katz is a librarian and former director of the Deaf Action Project at the Framingham Public Library, Framingham, Mass.

Alexander and Pendragon

A true story about children of the computer age

by Bruce Wellman

Computers have added a new dimension to the problems and challenges of teaching the exceptionally bright child. At any school that allows students access to computer terminals, some kids talk with ease about floppy disks and feedback loops while their teachers still struggle with BASIC at in-service workshops.

From time-to-time, *Classroom Computer News* will run profiles of these children of the computer age with the hope that insight into who they are and how they view and use computers will both raise questions educators need to ask when working with these students, and suggest some answers.

The Enzmann Brothers

Alexander and Pendragon Enzmann have no awe of computers. They've been working with them since their early teens, and, at ages 19 and 18 respectively, they are masters degree candidates in mathematics. "The computer is a marvelous toy," says Alexander. "It's a tool — like using a screwdriver," Pendragon adds nonchalantly.

The brothers live with their parents, Robert and Joanna Enzmann, and four sisters in a modest brick house on a quiet street in Lexington, Massachusetts. A salvaged dish antenna, propped unobtrusively next to a hemlock tree in the side yard, is the only exterior sign of the family's technological interests. The antenna is wired to one of the five television sets in the living room.

Four of the televisions are on when I arrive for a Friday night dinner. Alexander and his father are playing an intricate Japanese board game called Go, while they eat and talk with the rest of the family and answer my questions. The game is played with counters that are white on one side and black on the other. The board has 361 squares. "You can't program a computer to play Go like you can to play chess," Alexander tells me with a smile. "There are too many permutations."

Toward the end of the meal, the television output is reduced to the three network news broadcasts. Reports of the war between Iran and Iraq shift the dinner conversation to the topics of military hardware and desert weather patterns.

Dr. Enzmann interrupts the Go game to point out the sky color over a Middle Eastern city shown on one of the screens. His extensive service as a pilot during World War II and his doctoral degrees in geology, mathematical physics and electrical engineering make him a knowledgeable commentator. Mrs. Enzmann, a mathematician and computer analyst working on programs for large-scale Air Force radar, adds handily to the discussion of "backscattering" and "phased arrays," as jet fighters swoop across the screens.

Pendragon (the name is Welsh in origin) is pursuing a typical Enzmann academic career. After graduating from Lexington High School at age 14, he attended the University of Massachusetts in Amherst for two years and earned a bachelor of science degree in mathematics. He is now enrolled in a masters program in math at the University of Massachusetts in Boston. An additional year of study for a Ph.D. in math and two years in an accelerated medical school program will follow.

Alexander's academic career has been slowed a bit by the demands of competitive figure skating. Like his brother and his sister Heidi, who is a college freshman, he plans to join the Air Force and become a flight surgeon. "It's the only place in the world you can put your hands on heavy equipment — jets, trucks and radar," he says.

Pendragon had his first concentrated exposure to computers at age 13, when he began high school. "I learned all the common and uncommon languages — except for a few business languages, that first year," he says without a trace of cockiness. "I covered three years of computer language programs in three-quarters of a year. The programs were individually paced, and I could work on my own. There were no restrictions on computer use."

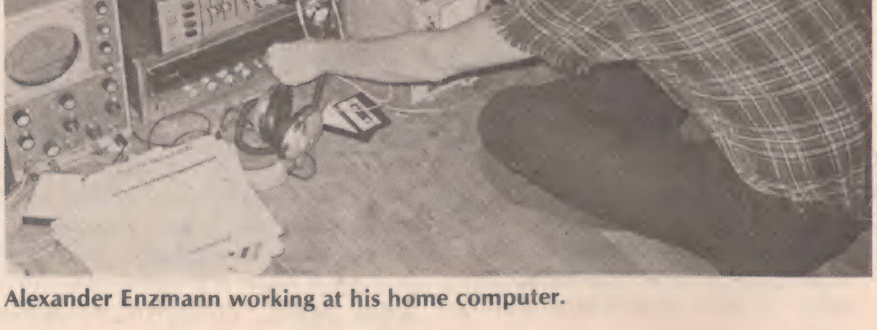
An Educational Philosophy

Dr. Enzmann believes that at a certain time developmentally, children can easily learn computer languages, just as they can master foreign languages. "If anyone has to take time to study any subject, there's something wrong," he says emphatically.

Pendragon agrees, proffering the family belief that studying inhibits learning and real creativity. "When the time is right, you can learn with incredible speed."

After rapidly mastering the languages,

Please turn to page 23.



Bruce Wellman

Alexander Enzmann working at his home computer.

Word Processors for Teachers

It's more than an intelligent typewriter — a word processor can help you get — and stay — organized.

by William D. Hedges

After six months of using a word processor, I would be extremely reluctant to go back to a typewriter for most writing tasks. One exception is the addressing of envelopes, but for most other duties the typewriter can perform, there is no comparison.

For example, for each of my classes I label one or more floppy diskettes. On each diskette I keep all lesson plans, quizzes, major examinations, bibliographies, class syllabus, names and addresses of students, handouts and even a file labeled IDEAS for teaching that subject.

By a word processor, I am referring to a computer program (software) that enables me to enter text; add, delete, rearrange, search through material; move blocks of text around; underscore, capitalize and correct mistakes, etc.

Advantages

Many high school, elementary and university teachers maintain and periodically update reading lists for their students. With my *Apple Writer* word processor, I can load the bibliography, insert new titles, delete old ones and print out an updated copy without having to retype the entire thing. This not only saves time, but also eliminates the costly errors that creep in through repeated typing of an entire list. In addition, I find myself much more willing to update my bibliographies than I was before. This goes for my lesson plans too. Working with a word processor is actually enjoyable!

Another advantage is that my tests get better and better. All faculty have experienced the frustration of test questions that were ambiguous. Students resent unclear test questions. Hence, in striving to be fair, teachers have to spend precious time correcting, giving additional credit, rerecording scores, etc. Rare is the examination that doesn't have some spelling errors, incorrect grammar and the like. With a text editor you simply go back to the original test, delete or modify the offending question and presto! For another time, you have a usable test.

Should you not want to reuse intact old tests, you can select some of the better items to incorporate in the new one. Also, you have a record of former testing and, at times, you will see ways to upgrade certain items to elicit a higher thinking level in students. And if you wish, you can insert a note or two of reminder to yourself about the test for future reference. It is virtually impossible for any of us to create a valid examination from scratch — another reason for periodic editing of the tests we do construct.

Because research evidence indicates that student study habits are affected by teacher testing habits, improvement of the quality of tests tends to bring about improvement in student study habits — a nontrivial side effect.

I am writing this article on a word processor. I will save a copy on diskette instead of messy carbon. The material's being on disk enables me to shift paragraphs about, delete or modify sentences, correct grammatical errors and the like without the necessity of retyping the entire copy.

For instruction, when material is the way you want it, sufficient copies for the entire class can be printed out. However, photocopying the additional copies is probably more practical. And look! No strikeovers and erasures!

Last, If you are like me and tend to lose materials and waste time searching for them, often without success, using a separate floppy disk for each class — or even several disks in some instances — means materials that work well are always there when you need them.

Disadvantages

Time is required to periodically update your materials; they are only as good as your input. The saying GIGO for garbage in - garbage out holds true here too.

You must have convenient access to a computer and a printer, plus safe storage space for your diskettes. Since schools by the hundreds are now acquiring microcomputers with printing attachments, the problem of access is rapidly disappearing.

The cost of the software is modest, ranging from fifty to several hundred dollars. Diskettes are about three to five dollars each and can contain considerable material. Other than the above few problems, I honestly can't think of any disadvantages.

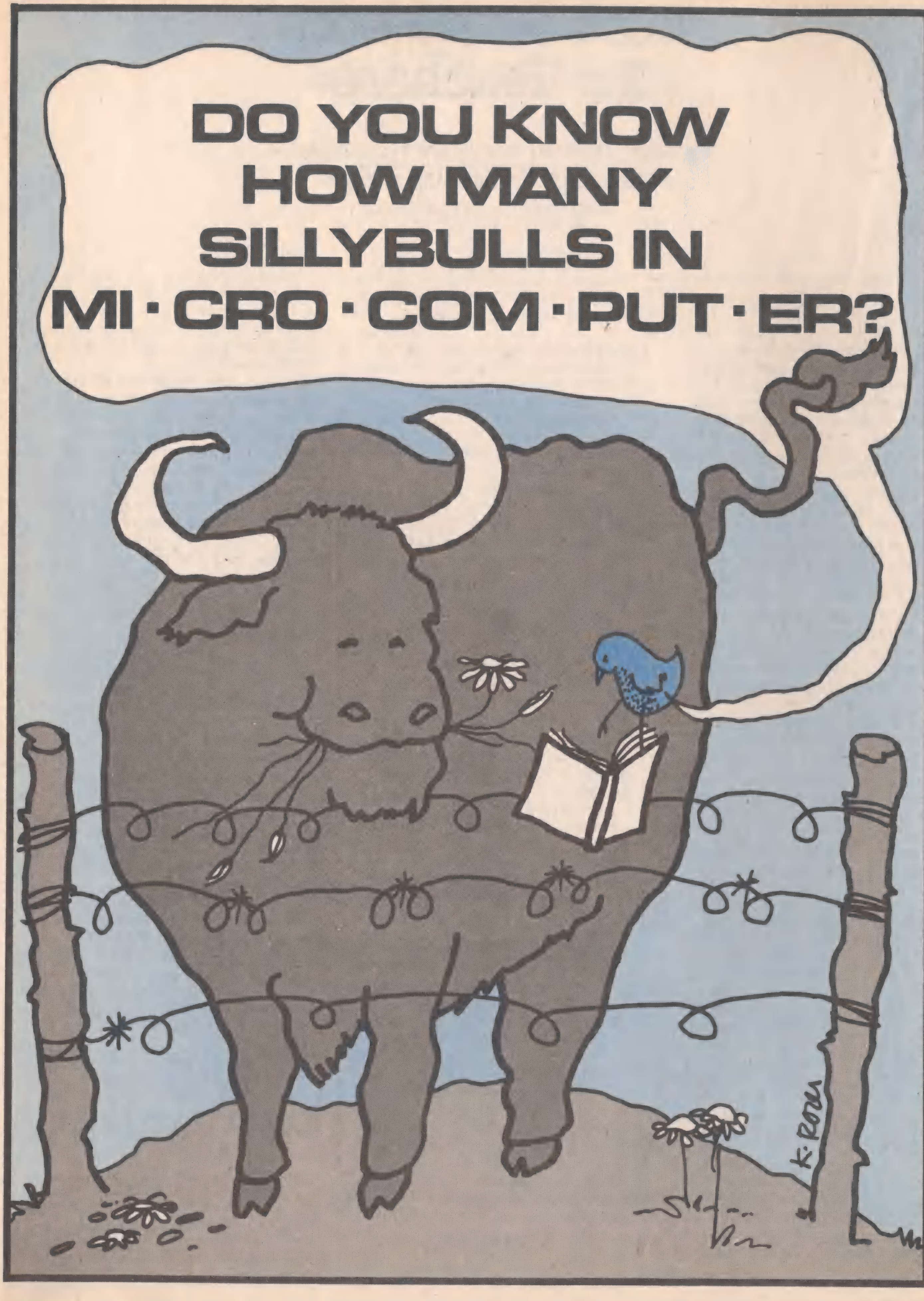
Choosing a System

In addition to the *Apple Writer* mentioned above, many word processors are on the market, and I do not intend to recommend one over the other. Such information can be best obtained from a review in one of the microcomputer journals. For example, *Super-Text* and *Easy Writer* are both reviewed in the July 1980 issue of *Creative Computing*. Others include *The Correspondent, Magic Wand, TRS-80 Mod II* word processing using the *WpDaisy, The Ed-80 Text Editor* and *WordPro*.

If you are planning to purchase a word processor, I advise you to read the reviews, make sure the software is compatible with your own or your school's system, and most certainly ascertain if what the word processor can do suits your needs. Every processor is different and some are far superior to others.

Last, of course, one can move over into the powerful (and expensive) business office systems with their multiple capabilities including word processing, but that's a subject for another article.

William D. Hedges is a long-time educator and educational writer, and a regional contributer for Classroom Computer News.


DO YOU KNOW
HOW MANY
SILLYBULLS IN
MI · CRO · COM · PUT · ER?
K. Rosa

Silly Bull

An udderly sensible syllabication game for young readers.

by Jeff Nilson

Most cattle enjoy the lazy days of farm life, but not Moses. He wanted to learn to read. Reading, he had heard, would let him visit the far corners of the world without ever leaving his field, and especially his favorite beech tree, under which he would lie on hot summer days and dream about France.

With a little practice, he learned to read short words like "Hill Top Farm," but he was still trying to understand big words like "registered holsteins."

"Can you help me figure out long words?" he muttered to his fodder one day.

"Why you silly bull," the old steer said. "All you need to do is break long words into their parts."

Good advice. Syllabication is one of the tools readers (including young bulls) use to figure out unfamiliar words. But as we learn as teenagers, breaking up can be hard to do. Children just learning syllabication must listen for syllables, learn vowel-consonant patterns, study prefixes, roots and suffixes, draw lines between syllables on worksheets, look for syllables in dictionaries and play syllable games.

Because Moses had a real stake in learning to carve up new words to figure them out, I designed a computer syllabication game called *Silly Bull* for him and for other students too.

How the Game Works

To win at *Silly Bull*, players must make more words than their opponents. Players move syllables on a four-by-four grid until they put the secret words the computer has chosen back together. A word is remade as long as its syllables are touching. The syllables can be in any order, and one point is scored for each letter in the word when it is remade. When the players find all the words, the player with the most points wins.

Before the game begins, the program selects some two or three syllable words which contain a total of 15 syllables. It randomly assigns the syllables to squares in the grid, leaving one square open. As the game progresses, the computer keeps track of each player's move.

Actually playing *Silly Bull* is as easy as if the game were on a board and players just moved syllable cards around. When they take their turns, players tell the computer which syllable they want moved into the empty space. The computer erases the syllable at the old location, and moves it to the empty space. If they have made a word, the players enter the word they've made. The program figures their score and prints it on the scoreboard on the right side of the screen.

The version of *Silly Bull* printed here contains 20 words taken from the Dale-Chall list. When you are ready to change the words, go to the part of line 15 that reads: DIM WL$(20,4). Change the "20" to the number of words in your list. Then change the commands that search WL$ to the number of words in your list. These are in lines 2210, 5465 and 6010. Replacing the DATA statements containing one word and its syllables is the last step before SAVEing *Silly Bull* with your word list in it.

Actually playing Silly Bull *is as easy as if the game were on a board and players just moved syllable cards around.*

Though *Silly Bull* does not actually teach syllabication rules, it does get reading students moving a lot of word parts around, sharing new words and using different strategies to outflank their opponents.

And Moses says, "There's no udder game quite like it. I loin more every time I play it. And I hope your students will make it their choice round after round."

Jeff Nilson, who teaches eighth-grade English in Dennis, Mass., is a contributing editor and roving pundit for Classroom Computer News.

The Silly Bull Program

(for the Apple computer)

```
]LIST

1  TEXT
2  HOME
10  DIM SL$(16,3)
15  DIM WL$(20,4)
20  DIM RS(150)
39  GOTO 110
40  FOR J = 1 TO 17 STEP 4
42  FOR I = 1 TO 25
45  HTAB I: VTAB J
48  PRINT "-"
50  NEXT I: NEXT J
55  FOR G = 1 TO 25 STEP 6
60  FOR H = 2 TO 16
65  HTAB G: VTAB H
70  PRINT "!"
80  NEXT H
90  NEXT G
99  RETURN
110  GOSUB 2000: GOSUB 2200
120  HOME : PRINT "'SILLY BULL' I
     S ABOUT TO BEGIN."
122  PRINT : PRINT
125  PRINT : PRINT "ENTER THE FIR
     ST PLAYER'S NAME."
130  INPUT "THEN PRESS 'RETURN'."
     ;P1$
140  PRINT : PRINT "ENTER THE SEC
     OND PLAYER'S NAME."
145  INPUT "THEN PRESS 'RETURN'."
     ;P2$
200  HOME : PRINT "PRESS THE NUMB
     ER OF CHOICE YOU WANT."
210  PRINT : INVERSE : PRINT "

           ": NORMAL
215  HTAB 6: VTAB 6: PRINT "1. DI
     RECTIONS"
220  HTAB 6: VTAB 8: PRINT "2. TO
      PLAY WITH THE GAME"
225  HTAB 6: VTAB 9: PRINT "   WO
     RDS SHOWING. (EASIEST)"
230  HTAB 6: VTAB 11: PRINT "3. T
     O PLAY AFTER SEEING THE WORD
     S"
235  HTAB 6: VTAB 12: PRINT "   F
     OR TEN SECONDS. (HARD)"
240  HTAB 6: VTAB 15: PRINT "4. T
     O PLAY WITHOUT SEEING"
245  HTAB 6: VTAB 16: PRINT "   T
     HE WORDS AT ALL. (VERY HARD)
     "
260  GET EA$
270  IF EA$ = "1" THEN  GOSUB 500
     0
280  IF EA$ = "2" THEN 350
290  IF EA$ = "3" THEN 350
```

```
300  IF EA$ = "4" THEN HOME : GOSUB
     3400: GOSUB 2400: GOSUB 40: GOSUB
     450: GOSUB 6200: GOTO 700
310  IF EA$ > "4" THEN 200
350  HOME : GOSUB 3400: GOSUB 240
     0
360  GOSUB 40: GOSUB 450
370  IF EA$ = "3" THEN  GOSUB 360
     0
385  GOSUB 6200
449  GOTO 500
450  FOR P = 1 TO 16
460  HTAB ( VAL (SL$(P,1)))
465  VTAB ( VAL (SL$(P,2)))
470  IF SL$(P,3) = "0" THEN  PRINT
     "      ": GOTO 490
480  PRINT ""SL$(P,3)""
490  NEXT P
495  RETURN
500  GOSUB 4400
700  REM  GAME MAINLINE ROUTINES
702  GOSUB 15000: GOSUB 6200: GOSUB
     6300
705  GOSUB 4400
710  PRINT "WHICH SYLLABLE DO YOU
     "
715  PRINT "WANT TO MOVE INTO"
720  PRINT "THE EMPTY SPACE?"
730  PRINT : PRINT "(TYPE IN THE
     SYLLABLE)"
740  INPUT "AND PRESS 'RETURN'.)"
     ;MW$
745  GOSUB 3000
750  GOSUB 4400
760  PRINT "HAVE YOU MADE A WORD?
     "
763  PRINT "(PRESS 'Y' OR 'N'"
766  INPUT "AND THEN 'RETURN')";P
     P$
770  IF PP$ = "Y" THEN  GOSUB 440
     0: GOTO 780
775  GOTO 700
780  PRINT "ENTER THE WORD,"
790  PRINT "THEN PRESS 'RETURN'."
800  INPUT HL$
820  GOSUB 6000
999  GOTO 700
2000  REM  POSITION ARRAY FOR SL$
2010  FOR I = 1 TO 16
2020  FOR J = 1 TO 3
2030  READ SL$(I,J)
2040  NEXT J
2050  NEXT I
2060  RETURN
2200  REM   SETS UP WORD AND
2205  REM   SYLLABLE ARRAY
2210  FOR I1 = 1 TO 20
2220  FOR J1 = 1 TO 4
2230  READ WL$(I1,J1)
2240  NEXT J1
2250  NEXT I1
2260  RETURN
2400  REM  PICKS SYLLABLES
2405  REM  AT RANDOM AND PUTS
2410  REM  THEM IN GAME GRID
2415  REM  AND SL$(16,3)
2420  REM  AT START OF NEW ROUND.
2425 RC = 1:S3 = 0:ST = 0:L2 = 0
2430 SR =  INT (3 *  RND (1)) + 1
2440  IF SR = 2 THEN 2430
2500  FOR Z3 = 1 TO SR
2510  GOSUB 5450
2517  IF WL$(RW,4) = "0" THEN RC =
      RC - 1: GOTO 2510
2520 S3 = S3 + 1:L2 = 4: GOSUB 25
     40
2522  IF EA$ = "4" THEN 2525
2523  GOSUB 3700
2525  NEXT Z3
2530  FOR ZB = 1 TO (15 - (SR * 3
      )) / 2
2533  GOSUB 5450
2534  IF WL$(RW,4) > "0" THEN RC =
      RC - 1: GOTO 2533
2535 L2 = 3: GOSUB 2540
2536  IF EA$ = "4" THEN 2538
2537  GOSUB 3800
2538  NEXT ZB
2539  RETURN
2540  FOR AZ = 2 TO L2
2545  GOSUB 3500
2550 SY$(AZ - 1) = WL$(RW,AZ)
2560  NEXT AZ
2570  REM  PUT EACH SYLLABLE IN S
      L$
2580  FOR AY = 1 TO (L2 - 1)
2590 RP =  INT (16 *  RND (1)) +
     1
2595  REM  CHECKS IF SPACE IS OCC
      UPIED
2600  IF SL$(RP,3) > "0" THEN 259
      0
2620 SL$(RP,3) = SY$(AY)
2640 ST = ST + 1
2650  NEXT AY
2670  RETURN
2900  IF WL$(RW,4) > "0" THEN 246
      0
2905  RETURN
2950  FOR AA = 1 TO 16
2955 SL$(AA,3) = "0"
2960  NEXT AA
2970  RETURN
3000  REM   MOVES MW$ INTO
3002  REM   EMPTY SPACE
3003  REM   ON SCREEN GRID AND
3006  REM  STORES NEW SYLLABLE
3008  REM  LOCATION IN SL$.
3010  FOR Q = 1 TO 16
3020  IF SL$(Q,3) = MW$ THEN 3040
3030  NEXT Q
3035  GOSUB 4200
3040 TW = Q
3050  HTAB ( VAL (SL$(Q,1)))
3060  VTAB ( VAL (SL$(Q,2)))
3065  PRINT "      "
3070  FOR R = 1 TO 16
3075  IF SL$(R,3) = "0" THEN 3090
3080  NEXT R
3090 TN = R
3100  HTAB ( VAL (SL$(R,1)))
3110  VTAB ( VAL (SL$(R,2)))
3120  PRINT ""MW$""
3130 SL$(TW,3) = "0"
3140 SL$(TN,3) = MW$
3150  RETURN
3400  GOSUB 4400
3410  PRINT "THE COMPUTER IS CHOO
      SING"
3415  PRINT "THE WORDS FOR THIS R
      OUND"
3420  PRINT "OF 'SILLY BULL'."
3430  PRINT : PRINT "PLEASE WAIT.
      "
3450  RETURN
3500 ZE = 16
3505  FOR CH = 1 TO 16
3510  IF SL$(CH,3) > "0" THEN ZE =
      ZE - 1
3512  NEXT CH
3515  IF ZE = 1 THEN 450
3520  RETURN
3600  REM
3605  FLASH
3610  HTAB 31: VTAB 13
3615  PRINT "YOU HAVE"
3620  HTAB 31: VTAB 14
3625  PRINT "10 SECONDS"
3630  HTAB 31: VTAB 15
3635  PRINT "BEFORE THE"
3640  HTAB 31: VTAB 16
3645  PRINT "WORDS"
3650  HTAB 31: VTAB 17
3655  PRINT "DISAPPEAR."
3660  FOR PP = 1 TO 6000: NEXT PP
3665  NORMAL
3670  GOSUB 4500
3680  RETURN
3700  REM  PRINTS GAME WORDS
3705  HTAB 27: VTAB 1: PRINT "THE
      WORDS FOR"
3710  HTAB 27: VTAB 2: PRINT "THI
      S GAME ARE"
3715  HTAB 27: VTAB 3: INVERSE : PR:
      "             ": NORMAL
3720  HTAB 29: VTAB 4 + Z3: PRINT
      ""WL$(RW,1)""
3750  RETURN
3800  REM
3820  HTAB 29: VTAB ((4 + Z3) + (
      ZB - 1)): PRINT ""WL$(RW,1)"
      "
3830  RETURN
3900  REM  PRINTS SYLLABLES
3904  REM  AS PART OF DIRECTIONS.
3910  HTAB 1: VTAB 1
3915  PRINT "AND THE SYLLABLES"
3917  PRINT "FROM THESE WORDS ARE
      "
3919  INVERSE : PRINT "
                 ": NORMAL
3920  FOR GG = 1 TO 16 STEP 2
3930  HTAB 3: VTAB 4 + (GG - 1): PRIN
      ""SL$(GG,3)""
3933  HTAB 11: VTAB 4 + (GG - 1):
      PRINT ""SL$(GG - 1,3)""
3935  FOR C = 1 TO 350: NEXT C
3940  NEXT GG
3950  RETURN
4000  HOME
4010  PRINT "DO YOU WANT TO SEE '
      SILLY BULL'S' WORDS"
4020  PRINT "BEFORE THE COMPUTER
      BREAKS THEM UP?"
4030  INPUT "(PRESS 'Y OR N')";EA
      $
4040  RETURN
4200  REM  GIVES ANOTHER CHANCE
4210  REM  IF SYLLABLE TO
4215  REM  BE MOVED IS NOT
4220  REM  CORRECTLY SPELLED.
```

```
4230  GOSUB 4400
4240  PRINT "THE SYLLABLE YOU"
4245  PRINT "ENTERED IS NOT SPELL
     ED"
4250  PRINT "CORRECTLY.  PLEASE E
     NTER"
4255  PRINT "THE SYLLABLE AGAIN. "
4280  PRINT "THEN PRESS 'RETURN'.
     "
4285  INPUT MW$
4290  GOSUB 3000
4295  GOTO 750
4299  RETURN
4400  FOR A9 = 1 TO 6
4410  HTAB 1: VTAB 17 + A9
4420  PRINT "
           "
4430  NEXT A9
4440  HTAB 1: VTAB 18
4450  RETURN
4500  REM  ERASES RIGHT SIDE
4505  FOR SX = 1 TO 17
4510  HTAB 27: VTAB SX
4520  PRINT "              "
4530  NEXT SX
4540  RETURN
5000  REM  PRINTS DIRECTIONS
5010  HOME : CALL  - 958
5020  PRINT "HERE IS HOW TO PLAY
     'SILLY BULL':"
5025  HTAB 5: VTAB 3: PRINT "THE
     COMPUTER FINDS SOME WORDS"
5040  HTAB 5: VTAB 4: PRINT "AND
     BREAKS THEM INTO THEIR PARTS
     "
5050  HTAB 5: VTAB 6: PRINT "EACH
     PART IS CALLED A SYLLABLE. "
5055 INVERSE : HTAB 27: VTAB 7: PRINT
     "      ": NORMAL
5056  PRINT : PRINT : PRINT "(PRE
     SS ANY KEY TO CONTINUE)"
5057  GET PP$
5058  HOME
5060 EA$ = "Y": GOSUB 2400: GOSUB
     3900
5075  HTAB 1: VTAB 19
5085  PRINT "THE COMPUTER PRINTS"
5090  PRINT "THESE SYLLABLES ON A
     GRID. "
5095  HTAB 1: VTAB 22
5100  PRINT "(PRESS ANY KEY TO CO
     NTINUE)"
5105  GET PP$
5106  FOR ZN = 1 TO 17
5107  HTAB 1: VTAB ZN: PRINT "
                              "
5108  NEXT ZN
5109  GOSUB 4400
5110  GOSUB 40: GOSUB 450
5120  HTAB 1: VTAB 18
5125  PRINT "PLAYERS MOVE THE SYL
     LABLES"
5130  PRINT "AROUND THE GRID UNTI
     L"
5135  PRINT "THEY MAKE A WORD. "
5138  FOR NG = 1 TO 10
5140 RX =  INT (10 *  RND (1)) +
     1
5145 MW$ = SL$(RX, 3)
5150  GOSUB 3000: FOR C = 1 TO 50
     0: NEXT C
5155  NEXT NG
5157  GOSUB 4500
5160  HTAB 31: VTAB 2: PRINT "A W
     ORD IS"
5165  HTAB 31: VTAB 3: PRINT "REM
     ADE IF"
5170  HTAB 31: VTAB 4: PRINT "ITS
     "
5175  HTAB 31: VTAB 5: PRINT "SYL
     LABLES"
5180  HTAB 31: VTAB 6: PRINT "TOU
     CH EACH"
5185  HTAB 31: VTAB 7: PRINT "OTH
     ER. "
5190  HTAB 31: VTAB 9: PRINT "THE
     Y DON'T"
5195  HTAB 31: VTAB 10: PRINT "HA
     VE TO"
5200  HTAB 31: VTAB 11: PRINT "BE
      IN THE"
5205  HTAB 31: VTAB 12: PRINT "CO
     RRECT "
5210  HTAB 31: VTAB 13: PRINT "OR
     DER. "
5215  GOSUB 4400
5220  PRINT "PLAYERS SCORE 1 POIN
     T"
5225  PRINT "FOR EACH LETTER IN A
      WORD"
5230  PRINT "THEY PUT BACK TOGETH
     ER. "
5235  PRINT : PRINT "(PRESS ANY K
     EY TO CONTINUE)"
5240  GET PP$
5243  RESTORE
5245  GOSUB 2000: GOSUB 2200: GOTO
     200
5250  RETURN
5450  REM  CHECKS THAT DIFFERENT
     WORDS ARE USED.
5465 RW =  INT (20 *  RND (1)) +
     1
5470  FOR X = 0 TO RC
5480  IF RW = RS(X) THEN 5465
5490  NEXT X
5500 RS(RC) = RW
5505 RC = RC + 1
5510  RETURN
6000  REM CHECKS IF THE PLAYERS W
     ORD IS CORRECT
6010  FOR AZ = 1 TO 20
6015  IF AZ = 21 THEN 6035
6020  IF WL$(AZ, 1) = HL$ THEN 607
     0
6030  IF  LEFT$ (HL$, 2) =  LEFT$
     (WL$(AZ, 1), 2) THEN 6050
6033  NEXT AZ
6035  GOSUB 4400: PRINT "YOUR WOR
     D IS NOT"
6040  PRINT "CORRECT.  IT IS THE"
6046  PRINT : PRINT "NEXT PLAYER'
     S TURN. "
6048 PRINT "(PRESS 'RETURN'": PRINT
     "TO CONTINUE)":
6049  GET PP$: GOSUB 15000: GOTO
     705
6050  GOSUB 4400: PRINT "IS YOUR
     WORD"
6053  PRINT ""WL$(AZ, 1)"?"
6056  PRINT "PLEASE ENTER YOUR"
6060  PRINT "WORD AGAIN. "
6065  INPUT HL$: GOTO 6000
6070  GOSUB 6400
6075  GOSUB 4400: PRINT "YOUR WOR
     D IS CORRECT. "
6080  PRINT "(PRESS ANY KEY TO CO
     NTINUE)"
6090  GET PP$
6099  RETURN
6200  REM
6203  HTAB 28: VTAB 15
6205  INVERSE : PRINT "   SCORE
     "
6207  NORMAL
6210  NORMAL
6220 HTAB 29: VTAB 16: PRINT "" LEFT$
     (P1$, 6)""
6270 HTAB 29: VTAB 18: PRINT "" LEFT$
     (P2$, 6)""
6300  REM  SHOWS WHOSE TURN IT IS
6305  IF PC = 2 THEN 6350
6310  IF PC = 1 THEN  FLASH
6320 HTAB 29: VTAB 16: PRINT "" LEFT$
     (P1$, 6)""
6330  NORMAL
6340  IF PC = 1 THEN  RETURN
6350  IF PC = 2 THEN  FLASH
6370 HTAB 29: VTAB 18: PRINT "" LEFT$
     (P2$, 6)""
6380  NORMAL
6390  RETURN
6400  REM  FIGURES SCORE
6410 LE =  LEN (HL$)
6420 SC = LE
6430  IF PL$ = P1$ THEN SQ(1) = S
     Q(1) + SC: GOTO 6460
6440  IF PL$ = P2$ THEN SQ(2) = S
     Q(2) + SC
6460  GOSUB 6700
6465 LE = 0: SC = 0
6470  RETURN
6700  REM  PRINTS SCORE
6725  IF PC = 2 THEN 6745
6735  HTAB 37: VTAB 16: PRINT SQ(
     1)
6737  IF PC = 1 THEN  RETURN
6745  HTAB 37: VTAB 18: PRINT SQ(
     2)
6760  RETURN
10000  DATA  2, 3, 0, 8, 3, 0, 14, 3, 0, 2
     0, 3, 0
10010  DATA  2, 7, 0, 8, 7, 0, 14, 7, 0, 2
     0, 7, 0
10020  DATA  2, 11, 0, 8, 11, 0, 14, 11,
     0, 20, 11, 0
10030  DATA  2, 15, 0, 8, 15, 0, 14, 15,
     0, 20, 15, 0
10100  REM
10110  REM  DATA FOR WORD AND
10120  REM  SYLLABLE LIST
10130  REM
10140  REM  CONFIGURATION: WORD,
10150  REM  1ST SYLLABLE, 2ND
10160  REM  SYLLABLE, 3RD SYLLABL
      E
10170  REM  OR '0' IF NO 3RD SYLL
```

Please turn to page 23.

Programming I: The Starting Gate

by Paul A. Shapiro

It's the first week of your computer programming elective. Your course outline overflows with important techniques, interesting classroom exercises and challenging lab projects. But where do you start?

In ten years of teaching introductory programming courses to high school students and continuing-education adults, we've found that no matter what their ability level or their age, students (and we'll have over 500 of them this year) respond well to the introductory activities described below. These exercises can help get your class off to a good start.

Activity I: How Do You Talk to a Computer?

Surprisingly, people of all ages still believe that the following simplistic paradigm is valid:

PROBLEM → COMPUTER → SOLUTION

In other words, they believe that you can take a problem, feed it into a computer and catch the solution as it falls out the other end.

During your first class, you may want to draw the above diagram on the blackboard and ask the class to help you make up lists of the types of problems computers can and cannot solve. Generally, the lists will look something like this:

Problems Computers Can Solve	Problems Computers Cannot Solve
Math Problems	Energy Crisis
Scientific Calculations	The Hostage Situation
Bookkeeping and Accounting	Inflation
Inventory	Social Problems

Since virtually everyone agrees that computers can solve math problems, you should ask *how* a user would enter a math problem into a computer, and you should suggest some possible command formats like the following:

1. Do the math problem at the bottom of page 12.
2. Please do the math problem at the bottom of page 12.
3. Solve for x: $4x^2 - 12x + 6 = 0$.

While everyone will agree that command one is ridiculous and that two is really no improvement, many students will feel that a command like three might be acceptable if phrased to the computer's liking.

The point you want to make, of course, is that computers can't convert raw problems into solutions, even when the problems lie in a computer-solvable domain and when they are phrased correctly. People must *program* computers. Undoubtedly someone in your class will volunteer this vital piece of information.

You want to lead the class discussion toward a more useful paradigm, such as this:

PROBLEM → ANALYSIS → PROCEDURE

→ TRANSLATION → PROGRAM

→ COMPUTER → SOLUTION

You want your students to understand that someone must perform two key steps in the process even before the computer makes an appearance: A human being must carefully *analyze* the problem and painstakingly devise a *procedure* for solving that problem, and a human being must *translate* that procedure into a format the computer can understand — namely a *program* written in BASIC, FORTRAN, Pascal or some other computer language.

At this point you should stress that your students themselves will be the key performers in both the analysis and translation phases of this process; the computer plays just a bit part.

As a sidelight, you may want to take an informal poll: "Which of the two functions — analysis or translation — do you think commands a higher salary in the job market?" Most students will correctly guess that analysts earn more than programmers. They typically predict that analysis involves more thinking.

This should lead you nicely to several important issues for your course:

1. Students will be programmer/analysts, involved intimately in both the analysis and translation phases.
2. Most people do consider the analysis phase more demanding, especially for beginners.
3. Analysts generally diagram the procedures they devise to make them easier to work with. The diagrams are called flow charts.
4. Beginners who wish to skip the flow chart phase and go directly from problem to program are effectively saying that they intend to skip the thinking phase.
5. Errors *(bugs)* will inevitably crop up in both phases. In analysis they are called *logical errors*, in translation, *syntactic (grammatical) errors.*
6. Logical bugs will be the harder of the two to find — many computers can locate syntactic errors for you, but none can detect logical errors in a procedure.

At this juncture, your students should realize that human beings have to do a lot of work before the electronic wizard gets into the act at all: "What good is a computer if I have to do all the thinking?" This question leads into your next activity — a discussion of the amazing speed and accuracy computers bring to the final phase of the problem-solving process.

Activity II: How Fast and How Accurate Are Computers?

Although Americans are much more sophisticated about computers than they were in the sixties or seventies, those who appreciate the speed at which computers operate are still in the minority. Typically, if you ask your class, "How fast can computers compute?" you'll get responses like, "Fast." "Pretty fast." "Very fast."

To help your students understand just what 'very fast' means, display a six-digit by six-digit multiplication problem on the blackboard. For example:

$$\begin{array}{r} 678432 \\ \times 956728 \\ \hline \end{array}$$

Ask your students to write this problem at the top of a blank sheet of paper and to work through the problem as quickly and accurately as possible when you say go.

Please turn to page 30.

Publishers *Continued from page 7*

correlate with standard math sequences.

The programs were originally written in Apple BASIC, but will also be available in Commodore PET and TRS-80 versions.

Bell & Howell

Bell & Howell has developed an English-based language called *Genis* that enables teachers with no previous programming experience to write their own software. The company has also developed a second, more sophisticated version of *Genis* for writing its own software. It plans to sell this version to other publishers interested in software development.

Separately, or in a package with *Genis*, Bell & Howell sells a version of the Apple computer which it has had modified for use in schools. The modifications include headsets for private use, volume control for whole class interaction, slots for external speakers, additional outlets and an audio mixer. In addition, the computer can control two monitors at the same time. Another unusual feature is a board controlled by software that lets the teacher interface existing films with the computer. The interface board will be available at the end of the year. Bell & Howell dealers provide maintenance service for the microcomputer.

Hayden Book Company

Hayden began developing software two years ago. Two of its packages are appropriate for educational use. *Crossbow* is a program for the Commodore PET that familiarizes elementary and middle-school children with fractions; *Programming in Apple Integer BASIC* is a 12-lesson programming course. It is also available in Applesoft.

Pre-engineering math programs for electronic technicians are another Hayden offering. While Hayden is presently acquiring programs developed by teachers and other experts in the field, it still intends to conduct additional marketing research to determine specific areas of concentration.

Houghton Mifflin

Houghton Mifflin produces *The Answer*, a hardware and software system that helps teachers with classroom management. *The Answer* can help with clerical work, with grouping students and with maintaining administrative records. SORBUS, a Pennsylvania-based company, provides service for the system.

Houghton Mifflin also produces *Stride*, a four-year old instructional system developed by Time Share Corporation (TSC), a Houghton subsidiary. *Stride* includes a reading program for grades three to six, a language arts program for grades four to six and a math program for grades one to eight. Teachers can use it for enrichment and remediation work as well as for regular classroom activities. *Stride* also includes a management system.

Stride is a time-sharing system that runs on a Hewlett Packard computer which TSC modified and sells. School systems buy the computer and then lease the programs on a yearly basis. Service can be provided either directly from Hewlett Packard or through TSC.

McGraw-Hill Book Company

McGraw-Hill has spent the past 18 months researching the educational software field, and it has developed several prototype programs that have been demonstrated for small groups of teachers, administrators and curriculum development specialists. Spokespeople are not yet willing to discuss the specifics of these programs, however.

McGraw-Hill is also screening acquired software (that which is developed by specialists in the field and submitted to McGraw-Hill for consideration).

Milliken Publishing Company

Milliken first released its *Math Sequences* package in October 1979 and offered a revised version a year later. The program runs on Apple, TRS-80 and Commodore PET microcomputers.

The math package covers 12 areas including number readiness, basic facts, laws of arithmetic, integers, decimals, fractions, percentages, equations and measurement formulas. It also includes classroom management routines.

New Milliken software will include a language arts package and an *Instructional/Communication Technology Reading Program*. The latter is intended for both remediation and regular instruction. It covers comprehension (middle school and high school levels) and word recognition (elementary school level).

Science Research Associates (SRA)

SRA has set up an Electronic Educational Products Department to coordinate software with existing SRA materials to aid authors in developing new products.

SRA is currently marketing three packages. *Computer Drill and Instruction: Mathematics*, intended for grades one through eight, is available both with and without a classroom management component. The management version helps teachers maintain student records and pinpoint the areas in which a given student is having trouble.

Fact Track, a package providing drill for math facts, and a class management system called *Classroom Management System: Mathematics* (grades 4-8) are SRA's other educational software offerings. In addition, many of their games have educational value.

While SRA has contracted to produce educational software for the Atari microcomputer, it also provides some for the Apple II Plus.

SRA has a toll-free number for educational software users experiencing problems with their software. Control Data Corporation takes care of any maintenance problems with the Atair microcomputer.

Scott, Foresman and Company

Scott, Foresman set up its Electronic Publishing Division last November. For this school year, the division produced the first section of a reading series called *Early Reading*. Also available is a school management applications program that aids in testing and classroom management. The company will present the first part of a math series this winter.

Scott, Foresman programs are designed to run on the Texas Instruments 99 microcomputer. They can be bought separately or as a package with the TI 99. Each software package is broken into units which are packaged at ROM packs. Texas Instruments will take care of any necessary hardware repairs.

— Phyllis Caputo

Forum *Continued from page 3.*

would better be spent on sturdier printers.

Research Needed.

The point is that at this time nobody really knows what communication features are best for each of the many areas of education. A great deal of experience and research is needed to define standards for computers that are optimal for education. And it is essential that such research be vigorously encouraged.

Meanwhile, we have in our hands right now powerful machines with the potential to engage the whole human being — mind, sensorium and body. If we view these computers as extensions of older media — as automated workbooks, for instance — we are seriously undervaluing a prodigious resource. Rather than arbitrarily censoring the communication capabilities that these machines do have to offer, we should be looking for the most creative and appropriate applications we can find to help us to make computer resources available to the whole person.

Microcomputers in the School Library's Future

by Katharine G. Cipolla

Few sources are available to librarians concerned with small systems automation or applications of microcomputer technology to library activities. Those that are, are both enthusiastic and expectant of the inevitable move of this technology into the library. In his article, "The Use of Microcomputers in Libraries" (*Journal of Library Automation*, March 1980), Allan D. Pratt concludes, "the microcomputer offers a challenge to librarians but also offers a significantly better and more effective way to carry out many library missions." Another recent report, *Microcomputers in Library Automation* (George Simpson, Mitre Corp., 1979) suggests that "micro-based systems appear to offer the best hope for automating libraries with small collections, such as school libraries." Yet neither of these pieces has much detail to offer.

This lack of specificity may be because few libraries have experimented with the new technology for automating library functions, as opposed to the larger number that have installed microcomputers for patron use. Even those libraries that may be using micros are not ready to report on their experiments. Other librarians can still only speculate about how microcomputers might best be applied to their library operations. So herewith, some speculations — both positive and negative — about the microcomputer's possible future role.

The Three Parts of Library Work

"Omniae bibliotecae in tres partes divisae sunt (all libraries are divided into three parts)," to paraphrase Caesar, and the greatest of these, in terms of public service, is information retrieval. The simplest and most common application of information retrieval in the small library is locating a book from cataloguing files. This is relatively easy using traditional card catalogues or book (or microform) catalogues. The human brain's ability to reason and the human eye's ability to scan make most catalogue data retrievable, regardless of the vagaries of the cataloguing system.

With a computer, this simple retrieval can be much more difficult. Programming and careful coding are all that stand between the patron and an irretrievable record. The literature abounds with anecdotes of minicomputer and main-frame computer systems with, essentially, "write-only" memories; microcomputers will not be immune from the problem.

Even more intricate is data retrieval from global information sources: the *New York Times* data base, ERIC, the U.S. Census, etc. For these, the amount of data is so vast that only the largest computers can handle it. Yet small and school libraries need the information. Happily, this is perhaps the easiest example of microcomputer application to library functions: the microcomputer becomes a terminal to access a larger computer over telephone lines. All it takes is a terminal program, a modem (an acoustic coupler with a cradle for the telephone receiver) linked to the micro's I/O port, and the telephone number and password of the larger computer. Sadly, line charges and access costs have so far limited access to large public, academic and research libraries. That microcomputer services like *The Source* are now offering time-sharing access to some of these files suggests that small libraries may soon be able to offer patrons on-line literature searches, too.

Few small libraries have yet experimented with the new technology for automating library functions.

Acquisitions and Circulation

The other two major areas of library activity are roughly analogous to management operations of commercial organizations: acquisitions and cataloguing are not unlike purchasing and taking inventory; circulation is very much like distribution and customer service. The real difference is the amount of data needed to run the business. While a shoe store, for instance, might use a microcomputer to keep track of an order for:

> "7 pr #122 Dexter loafers brown sz 6M vendor Q,"

a library will have to maintain order files like this:

> "2 c ISBN 0-89815-003-5
> Cuthbertson, T. 'Anybody's Bike Book' rev.ed. Ten speed pr.
> pbk bookjobber W."

The shoe store uses seven fields and less than 100 bytes. With the necessary indexing, perhaps 150 bytes per order would be needed. The library, on the other hand, needs eight fields and three times the number of bytes. With allowance for indexing, this order may need 500 bytes of storage. Furthermore, as the loafers progress into the store's inventory, their description will not change much; as the book is catalogued, it gains all the descriptors needed on the catalogue card:

> Cuthbertson, Tom
> Anybody's bike book, an original manual of bicycle repairs, newly revised & expanded ed., written by Tom Cuthbertson [and] illustrated by Rick Morrall. Berkeley CA: Ten Speed Press, c 1979.
> 200 p, illus, pbk ISBN 0-89815-003-5
> 1. Bicycles and tricycles — Maintenance and repair I. Morrall, Rick, illustrator II. title.
> TL430 .C87.

A large data file is now expanded to over 500 bytes and with indexing might use 700-800 bytes of storage space, not counting the expanded programming necessary to deal with all of this. For a very large system, like the Library of Congress's MARC (Machine Readable Cataloguing) system, the average is 1,500 bytes for a record and its indexing.

Similar, but less dramatic, comparisons can be made between customer service files and library circulation and borrower files. Although the functions are similar, the amount of data to be stored and manipulated is greater in the library.

Contingencies for Success

If, as the Mitre Corp. report suggests, the first application of microcomputer technology to libraries is the automation of existing procedures, success will depend on increasing storage capacity of the systems and on the availability of adaptable programming. When dependable high density storage devices, like the rigid disc or the high density floppy disc, are more generally available, a standard 32-64K microcomputer will have up to a million bytes of storage capacity. This should suffice for many operations of the small library. But, this added capacity increases the intricacy of the programing needed to manipulate the stored data. Currently, pre-tested commercial software for libraries is not available, and what is now being used for minicomputer systems has not yet been edited for microcomputer use.

As Pratt says, "The cost of pioneering in this area is not nearly so great as in [the case of other computer applications], nor is the penalty for errors quite so severe. Though it is true that pioneers frequently get shot full of arrows, it is also true that pioneers frequently lead the way to the future." Librarians can only hope that their intrepid colleagues soon begin to share their experiences,* their results and their programs.

*Please share your experiences with microcomputer applications in libraries. Send information to
Katharine G. Cipolla
c/o *Classroom Computer News*
Box 266, Cambridge, MA 02138

Katharine G. Cipolla is media services librarian at the Barker Library, Massachusetts Institute of Technology, Cambridge, Mass.

Silly Bull *Continued from page 19*

```
10200  DATA  MACHINE, MA, CHINE, 0
10210  DATA  MEASURE, MEA, SURE, 0
10220  DATA   OUTWARD, OUT, WARD, 0
10230  DATA  NIBBLE, NIB, BLE, 0
10240  DATA  REFUSE, RE, FUSE, 0
10250  DATA  PROMISE, PRO, MISE, 0
10260  DATA  RUBBISH, RUB, BISH, 0
10270  DATA  SETTLEMENT, SET, TLE, M
     ENT
10280  DATA  FORGOTTEN, FOR, GOT, TE
     N
10290  DATA  GASOLINE, GAS, O, LINE
10300  DATA  FLASHLIGHT, FLASH, LIG
     HT, 0
10310  DATA  HASTY, HAS, TY, 0
10320  DATA  DAYTIME, DAY, TIME, 0
10340  DATA  DOORSTEP, DOOR, STEP, 0

10350  DATA  ENGINEER, EN, GI, NEER
10360  DATA  ELDER, EL, DER, 0
10370  DATA  DONKEY, DON, KEY, 0
10380  DATA  CARELESS, CARE, LESS, 0
10390  DATA   BUTTERFLY, BUT, TER, F
     LY
10400  DATA  BROADCAST, BROAD, CAST
     , 0
15000  REM  WHOSE TURN
15010 PC = PC + 1
15020  IF PC = 3 THEN PC = 1
15030  IF PC = 1 THEN PL$ = P1$
15040  IF PC = 2 THEN PL$ = P2$
15050  RETURN
```

Computer Kids *Continued from page14*

the brothers used the computer as "mostly a plaything." "We would write programs that made the accumulator lights spell TILT," Alexander says with a grin. "We crashed the school system's computer a couple of times. Our program turned off all the inputs and no one else could get access."

The brothers and their friends soon learned how to gain entry to school and town records. "We never changed anything," says Pendragon. "We just thumbed through the attendance, finance and police records for fun. Once you know the access codes, you can get into any file."

Alexander extended the game beyond Lexington. This was possible because many computers are linked by the telephone system into a national and international net. "I had over 300 electronic traps searching for access numbers across the country. The programs automatically dialed phone numbers. If a human voice answered, the program moved on to another number until it found a computer." Using this strategy, he retrieved price and inventory statistics from a California meat market and got as far as Australia, where a computer sent him data on livestock herds and cattle futures.

Alexander and Pendragon felt they needed a deception to hide the game from instructors and computer center personnel. "The trick is to write a program that makes the terminal look like it's waiting for the next user," Alexander explains. "You log in under your own number and start the program, which makes the secretaries, or whoever is checking, think that you are done and the terminal is not in use. In reality, my program is running behind the screen."

Alexander is now a graduate assistant teaching a computer course at U.Mass. Boston.

The brothers see nothing unusual about their high school activities. "Some kids were into cars — gunning the engines and smoking cigarettes, and a group of us was into computers," says Pendragon. Their computer teachers were the only faculty members who knew about their talents and interests.

"If you're bright," says their father, "you have to hide yourself from adults. No child of ours will go near an advanced class — they're poison. Instead of one problem in geometry, they give you 50. We don't fight the system, we just ignore it. There is an implacable resistance to pushing ahead — you'll do it exactly their way or suffer. It's utter conformity."

Pendragon had to take an introductory programming course during his first year of college. "I slept through it," he says. His next computer course was on brain structure modeling.

Students in Control

Many schools have students who, like the Enzmann brothers, are exploring the uses of computers in a variety of sanctioned and unsanctioned ways. In some cases, these students know more about computers than do their teachers. Students who can program have a mastery of the computer and the data that is stored there. It is one of the few instances in school when students are in control instead of being controlled; it can change the balance of power in the school structure.

Educators need to consider the implications of having these bright and inquisitive youngsters in their classrooms. These students must be identified and productively challenged.

As teachers struggle to understand and apply computers in the schools, they must be aware of the students of the computer age who are not intimidated by the machines.

When I mentioned the lack of ease with the technology some adults are feeling, Pendragon Enzmann replied rhetorically, "Are you in awe of a light bulb?"

Bruce Wellman is an elementary school teacher and a free-lance photojournalist.

How Does the Computer Remember All That Stuff? Part 1

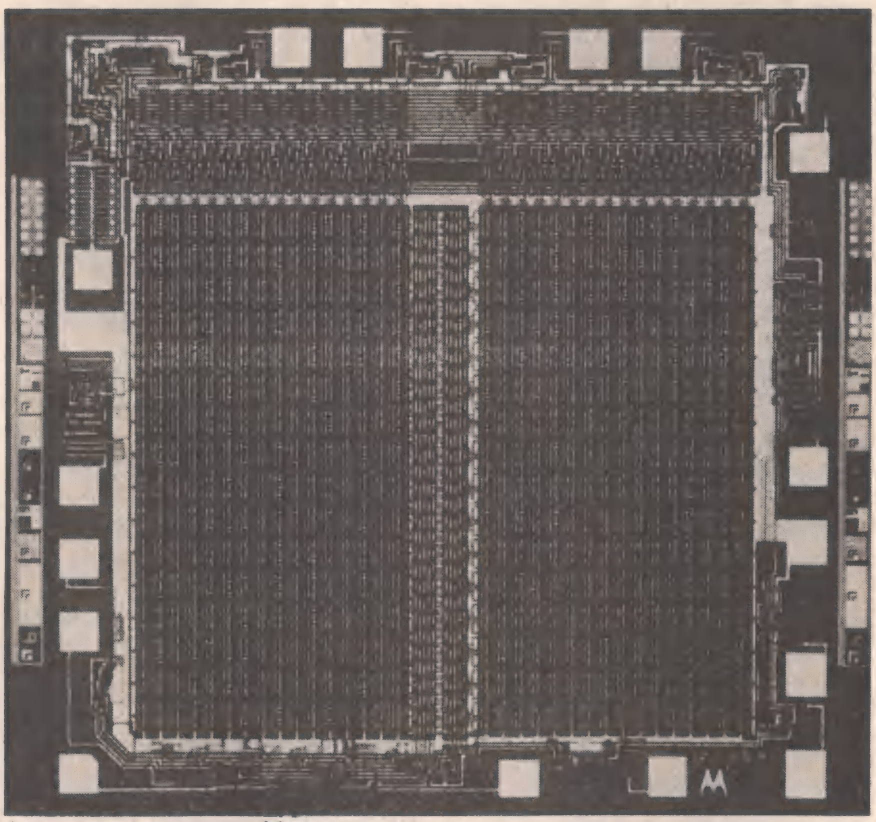
by Lloyd R. Prentice

Human memory, works more like a sieve than a bucket. The notion of using external mnemonic devices to plug holes in the grey matter goes back to, and probably beyond, some ancient accountant who needed to keep track of the headman's reindeer, or some prehistoric astronomer who wanted to keep the positions of the heavenly bodies straight as earth tilted through the seasons.

The simplest system employed one mark or symbol to represent one event — each notch on a stick stood for a single reindeer; the quantity of pebbles in a pile stood for the passing of days since the new moon.

But this system got out of hand when the numbers got up there. A prosperous accountant needed a cord of counting sticks to keep the king's accounts balanced; an accomplished astronomer contemplated entering the gravel business whenever he had to bring his pebbles in out of the rain.

For all its limitations, the one-mark-per-event system did establish a principle, and it led the way toward writing systems in general. Indeed, Francis Yates points out in *The Art of Memory* that the spread of literacy actually subverted the art of memorizing long speeches.



A computer's memory chip.

The Decimal System

A writing system is a memory aid *par excellence*. But the problem of denoting large numbers with simple representations remained a challenge long after writing systems found their niche in the market. In *One, Two, Three . . . Infinity*, George Gamow mentions primitive people who can't count beyond five. Everything beyond five is a heap. I suspect one reason behind the fall of Rome was the problem the generals had counting beyond MCXVIII.

Around 600 A.D. a very smart Hindu saw that if he could represent no number at all, or 'nothing', he could crack the large number problem. This was the thought behind the invention of zero, which, in retrospect, paved the way for the computer.

The reason the invention of zero shook the world was that it made possible positional number systems. A positional system based on ten symbols was invented in India shortly after the zero, was adopted by the Arabs around 800 A.D. and was picked up by European traders some 400 years later. This was the start of our decimal system. Given the decimal system, we can count until we run out of paper, patience or stamina and still not reach the highest number — and we need only ten marks to do so!

What do number systems have to do with how computers remember stuff? They make it easier to record distinctions, and distinctions are what memory is all about. When Og says he skinned 14 reindeer, he's thought to have better memory than Thor, who says he skinned a heap.

The First Computers

Not long after the decimal system caught on among European merchants, arithmetic became the rage in intellectual circles. Measuring, counting and calculating captivated some of Europe's finest minds. But counting to very large numbers was one thing, while multiplying two such numbers was another. A few visionaries saw possibilities in mechanical calculating machines and started tinkering in their basements.

Designers of early computers had a problem, however. A human could easily whittle the day's deer kill on a stick and read it back later, but how could a machine perform such a feat? The best that the early technicians

could think of were systems of gears and levers, but the results were less than thrilling. You have to push clockwork contrivances beyond the limits of mechanical precision to keep ten digits straight and to manipulate them in ways necessary to track profits in agribusiness or to plot a satellite trajectory. Not to mention that a mechanical computer is slower than a spavined horse on a gravel pile.

Early in this century, computer builders started experimenting with electronic devices. Such devices worked very fast, but as long as designers expected them to keep track of ten states, or digits, they tended to be cranky. But just as our clever Hindu showed that at times a small step back gains forward yardage, a clever computer scientist, Dr. John von Neumann, showed that a number system based on two digits — 0 and 1 — was more suitable for computers than the decimal system.

The binary numbering system fit hand-in-glove with electronics, and made computer memory possible.

Such a scheme, the binary system, had been kicking around for years, but it took a stroke of genius to see a practical use for it.

The Binary Solution

So what's the advantage of the binary system? Just try to find a physical representation of ten digits that's easy to manipulate. Fingers, beads on a wire, mechanical gears all do the job, but are dreadfully slow; as long as they were all that was available, progress in computers was nothing to write home about. The binary system fit hand-in-glove with electronics:

switch closed?	1
switch open?	0
voltage present?	1
voltage not present?	0
current flowing?	1
current not flowing?	0
light on?	1
light off?	0

Here we see a fine parallel between states of the physical system and digits in the binary system. Arrange a bank of switches in a row and you have an excellent analog for a binary number, where "off" corresponds to zero and "on" corresponds to one. A bank of eight switches can represent any one of 256 distinctions when you consider all the on/off possibilities. If you need more distinctions, just add switches. A bank of 16 switches can represent any one of 65,536 distinctions.

When you consider that binary numbers can do anything decimals can do, and that binary numbers can be used as well to represent letters of the alphabet, musical notes, dots on a video screen, or any number of other useful things, you begin to see the possibilities.

So, substitute the on/off state of an electrical switch for a notch on a stick, find a way to represent as many distinctions as possible with the fewest number of switches, work hard at making the switches small, cheap and easy to flick, and you come up with a modern computer memory. The marvel is that within the past two decades, engineers have discovered how to put more than 65,000 switches onto a single chip of silicon that's smaller than your fingernail. They organize each of these tiny switches into "bytes" or "words" (rows of eight or more switches) to form distinct memory locations, line the bytes up like so many post office boxes and use other switches to address each byte so that the computer can take information in or read it out. In so doing, they have created a mnemonic device that would make an ancient king's accountant green with envy.

CALENDAR

November

Microcomputers in Education. Nov. 18-20. Monroe Gutman Library, Harvard University, Cambridge, Mass. A Harvard University-sponsored conference focussing on learning theories and microcomputer software development. Speakers, publishing panel, electronic learning aids/game panel. Contact Microcomputer Conference II, Monroe C. Gutman Library, 6 Appian Way, Cambridge, MA 02138; 617/495-4225.

1980 New England Kindergarten Conference. Nov. 20-21. Lesley College, Cambridge, Mass. Covering topics such as "Computer Literacy Begins in Kindergarten"; also, a Computer Room. Contact The New England Kindergarten Conference, Lesley College, 29 Everett St., Cambridge, MA 02238.

The Northeast Computer Show. Nov. 20-23. Hynes Auditorium, Prudential Center, Boston, Mass. Software and hardware for all uses on display and for sale; also conferences on uses and evaluations of computers. Public invited; no preregistration necessary.

December

California Mathematics Council Annual Conference. Dec. 5-7. Asilmar State Conference Grounds, near Monterey, Calif. Program includes 12 computer sections, hands-on computer workshops, instruction on Apples, Ataris, PETs and TRS-80s; programs from the Computer-Using Educators software library will be available. Contact Bob McFarland, 685 A St., Hayward, CA 94541; 415/881-6196.

Computers in Elementary Education. Dec. 4-5. The Leamington, 1014 Third Ave. So., Minneapolis, Minn. Workshop for elementary educaors sponsored by the Association for Educational Data Systems; will focus on practical applications of computers in elementary classrooms. Contact Dr. Ken Brumbaugh, Minnesota Educational Computing Consortium, 2520 Broadway Drive, St. Paul, MN 55113;

January

Educational Software Symposium. Jan. 17-18. Holiday Inn, Bridgeport, Conn. Seminars covering topics from "Educational Software for Elementary Schools" to "Can Computers Teach English and Reading?" Early registration suggested. Contact Monica Kantrowitz, Queue, 5 Chapel Hill Drive, Fairfield, CT 06432.

Please send all calendar listings, two months in advance of the event, to: Calendar Editor, *Classroom Computer News*, PO Box 266, Cambridge, MA 02134.

other. In between are a set of skills, understandings, knowledge and values that play a large part in determining how well a person can function in society.

When I consider a person literate, I usually mean that she can read and write well and can use reading and writing fluently in many contexts and for many purposes. She regularly uses her literacy skills to satisfy personal needs, to conduct family affairs, to meet requirements of daily life and to meet the needs of her school, job, business or profession. Further, a literate person in our society will have had a range of experiences with the literature of our common cultural heritage. She will be able to read, understand, interpret and make judgments about a news article or a literary work, and will be able to compare the works of different writers.

On the other hand, I do not expect a literate person to be a professional writer, to be able to read fluently in a special field or even to spell perfectly all the time. I expect a literate person to have general command of language and literature, with all that that entails, but not necessarily to have achieved mastery of all language and literary skills.

A more complex aspect of literacy is its effect on a person's intellectual functioning. The ability to read and write can affect one's access to and storage of information, one's organization of ideas and even one's logical reasoning. All of these can affect daily functioning in profound ways. Simply put, a literate person can make use of a wider range of intellectual strategies than those available to someone who is non-literate.

How does someone grow up to be literate? I can suggest some major elements of the process. Literacy pervades our culture. Children almost absorb it by growing up in a society of literate adults. Older playmates, parents and teachers are all carriers of the culture of literacy. In school, all teachers — not just teachers of English — are presumed to be literate and to infuse literacy and the appreciation of literacy into all teacher/student interactions. From the earliest years of schooling, students both *use* literacy skills in a wide variety of everyday tasks of importance to them, and build new skills through teacher-directed activities.

Computer Literacy

With this understanding of what literacy means, I can describe computer literacy as that collection of skills, knowledge, understandings, values and relationships that allows a person to function comfortably as a productive citizen of a computer-oriented society. I have divided the concept of computer literacy into four distinct, but interrelated, areas:

1. The ability to control and program a computer to achieve a variety of personal, academic and professional goals. This includes the abilities to read, understand and modify existing computer programs, and to determine whether or not the program and/or the data it is using are correct and reliable.
2. The ability to use a variety of pre-programmed computer applications in personal, academic and professional contexts. This includes the abilities to make informed judgments as to the suitability of a particular software tool for a particular purpose, and to understand the assumptions, values and limitations inherent in a particular piece of software.
3. The ability to understand the growing economic, social and psychological impact of computers on individuals, on groups within our society and on society as a whole. This includes the recognition that computer applications embody particular social values and can have different impacts on different individuals and different segments of society. It includes the understanding necessary to play a serious role in the political process by which large and small scale decisions about computer use are made, and to transcend the dependent roles of consumer or victim.
4. The ability to make use of ideas from the world of computer programming and computer applications as part of an individual's collection of strategies for information retrieval, communication and problem solving. This aspect of computer literacy corresponds to the effect of learning to read and write on intellectual functioning and is probably the most difficult to incorporate specifically into educational programs, since the effects themselves are still not entirely clear.

Critical Ideas

As more and more schools begin to incorporate computer literacy into their programs and seek to use computers to enhance learning in other curriculum areas, I would like to raise certain considerations as critical for any computer-related programs.

Our understanding of what is meant by computer literacy will continue to expand rapidly. Observing the proliferation of computers and computer applications in our society, we are forced to recognize that the knowledge a competent citizen needs will expand with the increasing significance of computers. By introducing a particular set of computer literacy objectives into existing curricula, schools will not have *solved* the problem of computer literacy education. The infusion of computer literacy objectives into existing curricula is only a first step: school systems should be prepared for a process of continual expansion and redefinition of all computer-related programs during the next few years.

Who is in control of the computer may be one of the critical issues of the next few years. I believe that citizens of the 1980s and 1990s must understand the ways in which they can control computers and the ways in which computer systems and programs can be and are being used to control and manipulate *them*.

Children should learn to control the computer just as they learn to read and write — in the earliest grades.

A student's first encounter with a computer should put the student in charge. Many educators hold that computer programming is an "advanced" subject that we should save for high school or, in some cases, junior high school, while we expose younger students to a variety of pre-programmed instructional materials. I believe that precisely the opposite is true. Using child-appropriate computer languages such as LOGO, children should learn to control the computer, just as they learn to read and write in the earliest grades.

When students do interact with pre-programmed material, teachers should inform them of the justifications for such interaction and encourage them to evaluate both the software being used and the underlying rationale for using it. In this way, they will be developing a sense of personal understanding and control of computing on both the personal and societal levels. If students are to grow into intelligent users of computers, consumers of computer technologies and citizens capable of making decisions about computer use, all of these experiences must be built into their educational programs. We must design computer literacy activities that avoid unthinking acceptance or criticism of computer

applications.

Problems of Equity

Closely related to the issue of control is the problem of equity. Many educators have expressed the hope that the widespread use of computers in schools will enhance the education of those who have been excluded from the full benefits of our society. Unfortunately, certain trends now apparent may lead to exactly the reverse: a widening gap between a well-educated elite and a less-educated majority. In a society in which more and more jobs depend on computer-related skills, those who possess such skills will hold the most important positions. Those with a low level of computer literacy may find themselves increasingly relegated to menial or economically marginal positions.

Students from well-to-do families, or those who attend suburban or private schools, are already beginning to gain significant advantages in terms of learning to program computers and to use a variety of computer applications. Many families in suburban districts already have home computers, and the number of such families will continue to expand rapidly.

Schools may offer the only opportunity for many students to learn computer skills. It would be tragic if inner-city schools and schools in working class communities decide that they can't afford to purchase computers for their students or that the development of computer literacy is not the highest priority for use of the computers they do have. Ironically, many urban school districts have introduced computer-assisted instruction in an attempt to upgrade the basic skills of their students. This mode of computer use, in which the computer tells the student what to do, seems ideally suited to producing "second-class citizens" in a computer-based society.

A related concern is the domination by male students and teachers of many educational computing situations. Many high schools report that equal numbers of males and females take introductory computing courses, but that advanced courses and computer clubs are predominantly male. With computers and computer skills assuming increasing social importance, the danger that computing activities may reinforce male/female role stereotypes must be explicitly recognized and positively dealt with in curriculum materials, teacher training and program implementation.

Computer Literacy for Teachers

It is hard to imagine students becoming computer literate without computer literate teachers. When we understand how children growing up in a literate culture acquire the knowledge, skills, understanding and values that the conventional concept of literacy encompasses, we recognize that developing a computer literate school staff is a major aspect of education for computer literacy. The inclusion of computer literacy units or courses in the existing curriculum tends to bypass the long-range issue of staff development. We should see such a program as merely a first step — ultimately inadequate if not part of a long-term growth and development effort.

Educators face yet another question of priorities. Universal computer literacy is a basic skill of the 1980s and deserves a major role in the school curriculum. To educate students for computer literacy, schools must develop leadership, curricula and computer-literate teaching staffs, while acquiring the necessary hardware and software. While many aspects of this development will be intangible, two tangible aspects will undoubtedly be money and time.

Computer literacy will cost money. Equipment, software, curriculum, leadership development and teacher training will all require substantial investment. The money is not likely to come from a major increase in local budgets, or from state or federal largess. Rather, it must come from a reallocation of a diminishing pool of funds at the local level.

Computer literacy will take time. It takes a minimum of about one year to train a resource teacher who can begin to provide real leadership. It will require a substantial commitment of in-service training time for those teachers who have major responsibilities in this area, and some in-service training time for all teachers. Like the money needed to support computer education, time will have to be reallocated from other areas.

Redefining Local Priorities

The responsibility for computer literacy education is spread throughout many levels of our educational system. Federal programs, which are likely to increase somewhat during the next few years, have provided some support for research and development. Some states have formed consortia to centralize planning, curriculum development, teacher training and equipment acquisition. A few schools of education have begun embryonic graduate programs in computer education. Some computer manufacturers have offered substantial discounts to schools in their local areas, thereby supporting the acquisition of equipment.

In concentrated high-tech areas, industry councils are lobbying for and supporting a variety of technical education programs. Textbook publishers, in alliance with microcomputer manufacturers, are beginning to develop computer-related programs, although these are mostly oriented toward computer-assisted instruction rather than computer literacy. Finally, we have seen national and regional educators' groups emerge to facilitate the exchange of information and ideas.

Support from outside the schools will remain a minor factor without a major redefinition of local priorities. I believe that whether or not schools will be a serious factor in educating a computer literate public by 1990 is an open question. The failure of schools to make a major commitment in this area now can have disasterous consequences for both the education of the public and the future of public education.

Dr. Daniel H. Watt is director of the Computer Resource Center at Technical Education Research Centers in Cambridge, Mass. He is also research associate with the MIT LOGO group and LOGO project coordinator for the Brookline Public Schools.

Products briefs are written from manufacturers' announcements.

Micro-PLATO

Control Data Corporation has launched a new, lower-cost computer-based education system called Micro-PLATO. Micro-PLATO, a derivative of the company's PLATO time-sharing educational system, runs on a standard PLATO terminal (the Information System Terminal II with 32K RAM) and a Master Flexible Disk Drive.

The new system capitalizes on the microprocessor chip in each PLATO terminal to control and channel lesson materials from the disks to the terminal screen. Because lessons can be delivered without connection to a central host computer, the new delivery method can result in significant cost reduction.

Control Data Corporation's Mathematics Series and Reading Series will be the first Micro-PLATO programs, available in the first quarter of 1981. For more information, write Control Data Corporation, PO Box O, Minneapolis, MN.

Courtesy of Control Data Corporation

Control Data Corporation's Micro-PLATO system delivers PLATO lessons independent of a central host computer.

Computer-Aided Calculus

Microphys Programs has recently released a series of 12 programs designed for use in introductory calculus courses on both high school and college levels. The programs cover: differentiation and integration of both algebraic and trigonometric functions; maxima/minima problems; integrations of areas of plane figures, of volumes of solids, of arc lengths and of surface areas of solids.

Every program generates a unique set of problems, with or without accompanying answers, for each student. The student can complete the problems away from the computer. When finished, he or she enters a personal code number and the answers to the problems into the computer. The computer will grade the work, displaying answers to those questions which were incorrectly solved; it also displays a percent score and a brief, overall evaluation.

The calculus programs are intended for use with a Commodore PET/CBM microcomputer having at least 8K of storage. Individual programs are available on cassette; all 12 are available on a diskette. Both versions include instructions.

For further information, contact Microphys Programs, 2048 Ford St., Brooklyn, NY 11229; 212/646-0140.

CAI for the TRS-80

CAIWARE, a software system for writing and administering true-false, multiple choice and completion-type lessons, is now available from MicroGnome. Developed for Radio Shack's 16K TRS-80 microcomputer with Level II BASIC, CAIWARE presents a set of prototype questions that guide teachers in programming their own lessons. Once the teacher has programmed the lesson, students can run through it on their own. The program tabulates students' scores for each topic area. The teacher can design the program so that, if a student's score is high enough in a particular topic sequence, the computer automatically skips to a more advanced lesson. Similarly, the program can return to more elementary lessons as needed.

For more information, contact MicroGnome, 5843 Montgomery Road, Elkridge, MD 21227.

Stars in Your Apple

Tellstar, a high resolution graphics astronomy program, identifies, locates and provides information on stellar objects. *Tellstar* will run on Apple microcomputers with 48K memories, Applesoft and at least one disk drive.

Tellstar lets the user see what the skies look like over the North Pole, over a South Sea island — over any location on earth — from eight different directions, as well as straight overhead. With single letter commands, users can connect constellations or locate objects listed on star tables. The program accurately adjusts for latitude and longitude, for viewing time and date, and for the earth's precession (wobble).

For more information, contact Scharf Software, PO Box 18445, Irvine, CA 92713; 714/557-9206.

For the Administrator

Systems Unlimited has released SUNPAC, an administrative software package intended for use in elementary and secondary schools and community colleges.

SUNPAC is written in ANSI '74 COBOL and can operate on mini- and microcomputers, as well as on main frames. Although designed primarily to operate on Data General equipment, SUNPAC is not restricted to one manufacturer.

The SUNPAC software contains systems for payroll, budgetary finance, student records, grade reporting and student scheduling, among others. All systems may be used separately or together; they are distributed on diskettes, hard disks or magnetic tape.

Contact Systems Unlimited, Inc., Suite A, Storey Building, Town & Country Shopping Center, Aberdeen, NC 28315; 919/944-7022.

Ohio Scientific recently developed a desk top business planner, *MDMS Planner*, designed for exclusive use on Ohio Scientific's line of personal and business computers.

MDMS Planner is designed for anyone working with numbers. Administrators can use the planner to handle budget planning, capital budget planning and cash flow planning, and can easily make changes or comparisons using the MDMS "What

If" recalculation feature.

Unlimited model size, formatted report generation and English-like calculation rules are functions unique to *MDMS Planner*. Another feature is a plotting capability which allows a user to plot results on a bar, cumulative bar or point graph.

For more information and a demonstration, call your local Ohio Scientific dealer or call toll free 1-800/321-6850.

The CONSTELLATION from Corvus

The Corvus CONSTELLATION enables two to 64 personal computers to share one high-density storage disk. Called a *multiplexer*, the CONSTELLATION makes up to 40 million bytes of Corvus hard disk capacity available to the personal computers. (That's the equivalent of the amount of information stored on 29,000 typed double-spaced manuscript pages.) In addition, computers in a CONSTELLATION network can share peripherals, such as a common printer, and can communicate in a fully interactive mode.

The CONSTELLATION is a network in which multiple computers are connected in a star configuration. The center of the star is the Corvus CONSTELLATION host multiplexer. This central node contains hardware that polls up to eight computers in a round robin fashion. A two-level network containing up to eight host multiplexers connected to a central multiplexer allows up to 64 computers to share the disk.

Any personal computer compatible with the standard Corvus disk system is compatible with the CONSTELLATION.

For more information, contact CORVUS SYSTEMS, Inc., 2029 O'Toole Ave., San Jose, CA 95131; 408/946-7700.

Courtesy of Personal Software

Personal Software Inc.'s *VitaFacts Series* includes the six health education programs pictured above.

Health Education

Six health education programs are now available from Personal Software Inc. for Apple II, Commodore PET/CBM and TRS-80 personal computers.

The *VitaFacts™ Series* includes packages named "Growing Up," "Talking About Sex," "Drinking and Drugs," "Birth Control," "Your Blood Pressure" and "Heart Attacks." Created by two Canadian firms — Medifacts Limited and the Richmond Software Group — the six programs have the approval of the Canadian College of Family Physicians.

The software packages are designed for use by families, schools, health education organizations and medical professionals.

Each package comes with a 12- or 16-page instruction manual containing an introduction to the subject, illustrations and diagrams, a glossary of medical terms and complete, step-by-step loading and operating instructions for each of the three computers.

The *VitaFacts Series* is the first educational software offering from Personal Software. For more information, contact Personal Software, Inc., 1330 Bordeaux Drive, Sunnyvale, CA 94086; 408/745-7841.

Super Text for Apple

MUSE Software is now marketing *Super Text*, word processing software for the Apple microcomputer.

With *Super Text*, Apple users can enter files into memory, print them and save them on disks. The files can be reloaded for editing, appending, changing or printing. Users can also summon any file on any disk with a single command; insert and delete text from the files with a floating cursor; scroll forward and backward either a line or a page at a time; and jump to either the beginning or the end of text. A split screen allows users to edit and reference material at the same time.

Super Text can also operate in a Math Mode — a 15-digit, on-screen calculator that employs both an item count and an accumulator. Both direct calculations and calculations on numbers in the text are possible.

Super Text is available through: MUSE Software, 330 N. Charles St., Baltimore, MD 21201; 301-659-7212.

Programs for Teaching Apple BASIC

Charles Mann & Associates' Micro Software Division is now marketing three self-teaching programs for Apple BASIC: The Basic Teacher, Floating Point Dictionary and Basic Teaching Pac.

The Basic Teacher is a 13 lesson, conversational program referenced to Apple's BASIC language and system operation manual. Lessons 1-12 teach Integer BASIC. Tests, sample programs, review sections and a general index of program contents are all provided. The last lesson is an Apple Tricks Program. This is a sound and graphics program that explains and tests the peaks, pokes and calls used to produce graphic and sound effects.

Floating Point Dictionary (available for disk only) is both a teaching tool (for Applesoft II and BASIC Programming Language) and a reference system useful to both beginners and experienced programmers. The system allows one to ask for a definition of any BASIC term. In response, the user will get a definition of that term, working examples of statement forms and a test program which can be run to show the results of the command's use. A "Help" command aids baffled users in locating the right command.

The Basic Teaching Pac is a double disk package for teaching both of Apple's BASIC languages. Over 150 KB of program from The Basic Teacher and Floating Point Dictionary are included.

For additional information, contact any Charles Mann & Associates dealer, or write or call Charles Mann & Associates, Micro Software Division, 7594 San Remo Trail, Yucca Valley, CA 92284; 714/365-9718.

BOOK REVIEWS

Are You Computer Literate?
by Karen Billings and David Moursund
Dilithium Press, Portland, Ore., 1979
(Paperback)

A computer literate person understands what computers can and cannot do, how they are used in society and how they may actually change our lives. This book introduces the beginner to these three components of computer literacy.

The authors have identified eight topics which form the major chapters:

- What Is a Computer?
- Why Do Computers Exist?
- Data-Entry and Computer Programming
- Smart Machines
- How Computers Are Being Used
- How Computers Affect People
- What Else Is There to Know?

Each starts with a true-false quiz that helps the reader begin thinking about the content of the chapter. The discussion that follows is short and is designed to provide a general overview of the topic. A variety of different activities that can be used to refine or extend the reader's knowledge conclude each chapter.

Throughout, the authors relate the content to the reader's own background and experience. This is particularly evident in the activities following the chapters. These activities are both informative and fun to do. For example, one activity following a discussion of data entry has the reader involved in alphabetically ordering a set of fifty words (each is on an index card). Following this, the reader is directed to write a detailed set of instructions for completing this task. These instructions should be simple enough for a third- or fourth-grader to follow. Such an activity could lead to a class discussion of the need to be explicit in writing such instructions. An interesting extension would be to test the instructions with a third-or fourth-grade child.

A final chapter provides a very good list of additional resources such as books, films and records. It is followed by a computer literacy test (multiple choice) and a glossary of terms.

This book would be an excellent choice for intermediate level through high school grades. It is one of the few available for use with students at this level that is both appropriate *and* very well done. Students will find the presentation of topics easy to read and to understand and will get involved in the follow-up activities. They will be computer literate following their work with this book. — **Susan Freil**

Susan Freil, a math specialist, teaches at Lesley College, Cambridge, Mass.

LETTERS

I certainly enjoyed reading the first issue of *Classroom Computer News* as I appreciated your professional approach to such an important activity. Keep up the good work!

Bodie Marx, Vice President
Milliken Publishing Company
St. Louis, Missouri

Just received Volumn 1, No. 1 of *Classroom Computer News*. Congratulations upon launching what appears to be an excellent service with enormous potential.

I'll look forward to future issues of *Classroom Computer News*.

Harry Wolford, Director
Division of Computer Services
and Statistical Reports
Department of Education
Columbus, Ohio

Congratulation on a wonderful first issue. The articles are up to date and right on the mark for computer educators. As I have put out our own newsletter for two years, I have had the dream that someone else would *do it right*, and you have. Thanks for the quality effort. You obviously understand the problems and possibilities of computers in education.

William J. Wagner, President
Computer-Using Educators
Mountain View, California

I received your first issue and I'm glad I was on your mailing list. It's packed with down to earth information that should really fill a void for those of us within the schools. From the *Forum* to the *Book Reviews* there was something of value and interest.

I'm spreading the word and wish to add my vote of confidence and congratulations.

Robert C. Conroy, Principal
C.T. Douglass School
Acton, Massachusetts

Letters to the Editor should be sent to:
Letters, *Classroom Computer News*,
Box 266, Cambridge, MA 02138.
Letters may be edited for length and clarity.

Math/Science

Continued from page 20.

Have them raise their hands as they finish. Using a clock or a stopwatch, keep a scoreboard. Usually, completion times range from three to seven minutes for students with average-to-good mathematical ability. The average is generally five minutes.

When all the students have finished, walk around the room reading aloud their answers. Ask for matching answers from the audience as you read each one. In a typical class, very few (if any) answers will match. Here's your chance to make a point about computer accuracy vs. human accuracy.

Finally, you want to contrast human and computer speeds. Ask how long your class thinks a computer would take to solve the multiplication problem. Responses may range from 10 seconds to one-trillionth of a second, but students tend to offer conservative guesses.

The actual speed will vary dependingg upon the size and cost of the computer system, but modern electronic computers typically handle problems of this type in times on the order of a millionth of a second — i.e., computers can do roughly one million difficult multiplication problems in a second.

Since few of us can fathom a million events happening in a second, the following exercise is a tremendous aid: have your students calculate how long each of them would take to solve a million multiplication problems at their speed.

Assuming an average of five minutes per problem, you can make a work chart like this:

Time-frame	Number of Problems
5 minutes	**1**
1 hour	**12**
1 work-day (8 hours)	**100****
1 work-week (5 days)*	**500**
1 work-year (50 weeks)*	**25,000**
1 career (40 years)	**1,000,000**

In other words, the computer can do in *one second* what one good math student could accomplish in an entire career!

This incredible speed, coupled with the computer's nearly unfailing accuracy, makes the computer a force worth studying and mastering. May the force be with you.

*The boss gives us weekends off and two weeks rest during the summer.

**rounded off from 96.

Paul A. Shapiro is coordinator of educational computing services for the Newton Public Schools, Newton, Mass.

COMPUTER IN YOUR FUTURE?

Classroom Computer News

TELLS YOU WHAT YOU WANT TO KNOW IN LANGUAGE YOU UNDERSTAND

The classroom computer is here and it's the most exciting educational innovation ever.

And **Classroom Computer News** is here to help you stay abreast of what's new in educationally significant computer applications, products, research, grants and governmental programs.

Classroom Computer News speaks your language.

It's written by teachers and administrators for teachers and administrators. Each issue is packed with news, features, how-to's, profiles and reviews — all written and edited to help you get the most out of computers in your school, from kindergarten to graduate school, from sight words to vocational education.

Classroom Computer News

NOVEMBER-DECEMBER, 1980

VOLUME 1, NO. 2

$2.00

MICRONEWS

• The Japanese are coming . . . with low-cost, high-capacity disk drives that portend impressive storage capacities for future education-oriented computers. The drives, introduced by Fujitsu, a $2.5 billion company, store 84 megabytes and cost $8795. This is 16 times the amount of information in Webster's Seventh New Collegiate Dictionary.

• Wang Laboratories, major manufacturer of small computers, has had problems locating programmers so it's started its own graduate school. The Wang Institute of Graduate Studies, Tyngsborough, Mass., opened recently to applicants from across the country. The Institute, says Dean Caroline Wardle, plans "to remain small and select." Curriculum blends academic, management and technical subjects.

• PLATO, well-known educational software package for time-sharing systems, may soon be available for Apple computers. Apple's Glen Polin says several people are now converting PLATO programs. This, he feels, will greatly expand the educational software available for the Apple.

CONTENTS

Bruce Wellman

Computer literacy — for both students and teachers — will be the key to successful educational use of computers during the 1980s.

Computer Literacy: What Should Schools Be Doing About It?

by Daniel H. Watt

The term computer literacy is coming into widespread use without a clear, generally agreed upon definition. I think that the concept remains poorly defined for several reasons. First, the everyday concept of literacy is not clearly defined, having different meanings in different contexts. Second, computer technology is proliferating so rapidly in all phases of our lives that any particular notion of the specific knowledge necessary to be computer literate tends to be surpassed as quickly as it is formulated. Third, now that there is widespread agreement that computer literacy is of fundamental national importance, many people with diverse goals are trying to attach that label to their own particular ideas. Yet we need to formulate a concept of computer literacy that we can use to develop the new educational programs we will require during the next few years.

Defining Literacy

My definition of computer literacy is based on an interpretation of the common meaning of literacy. Dictionary definitions of literacy usually include phrases such as "the ability to read and write" and "the state of being well informed, educated." These definitions include no specific standards. The concept of what it means to be literate can vary from culture to culture, and from group to group within a given culture. Literacy has to be viewed as a continuum, from a minimal ability to read news headlines at one extreme to the literacy skills of a professional writer or skilled academic at the

Please turn to page 26.

Classroom Computer News, Box 266, Cambridge, Massachusetts 02138.

SUBSCRIBE TODAY!

One year (six issues) . . . $9.00

Classroom Computer News

☐ My check is enclosed

☐ Please bill me $9.00 for a one-year subscription

Name

School

Position

Address

City State Zip

Classroom Computer News
P.O. Box 266
Cambridge, MA 02138

NONPROFIT ORG.
U.S. POSTAGE
PAID
BOSTON, MA
PERMIT NO. 3519